

# UNIVERSITÉ BLAISE PASCAL – CLERMONT II

Ecole Doctorale Sciences Pour l'Ingénieur

## Thèse Préparée au LASMEA

UMR 6602 CNRS / Université Blaise Pascal  
(Laboratoire des Sciences des Matériaux pour l'Electronique, et d'Automatique)

Spécialité : Vision pour la Robotique

Présentée et soutenue publiquement par

**François Bardet**

le 30 octobre 2009

## **Suivi et Catégorisation Multi-Objets par Vision Artificielle Applications au Suivi de Personnes et de Véhicules**

Devant le jury composé de

Directeur :	Michel Naranjo	Professeur Émérite	LASMEA
Co-Directeur :	Thierry Chateau	Maître de Conférences	LASMEA
Rapporteurs :	Frédéric Lerasle	Maître de Conférences HDR	LAAS
	Jean-Marc Odobez	Senior Researcher	IDIAP
Examineurs :	Jean-Thierry Lapresté	Professeur	LASMEA
	Philippe Lepert	Directeur de Recherches	LCPC
	Patrick Sayd	Chercheur	CEA



# Table des matières

<b>1</b>	<b>Suivi d’Objets : Panorama</b>	<b>1</b>
1.1	Vue d’ensemble . . . . .	2
1.2	Typologies des systèmes de suivi d’objets . . . . .	4
1.3	Méthodes d’inférence d’état pour le suivi d’objets . . . . .	5
1.4	Inférence par résolution directe . . . . .	7
1.4.1	Inférence d’État Déterministe . . . . .	7
1.4.2	Inférence d’État Probabiliste . . . . .	7
1.5	Rapport au temps . . . . .	7
1.5.1	Détection d’état instantané . . . . .	8
1.5.2	Estimation de trajectoire <i>a posteriori</i> . . . . .	9
1.5.3	Suivi d’État Causal . . . . .	10
1.5.4	Suivi d’État Causal Séquentiel . . . . .	10
1.6	Représentation des Lois de Probabilités pour le Suivi Probabiliste . . . . .	11
1.6.1	Suivi Probabiliste à Représentation Paramétrique . . . . .	11
1.6.2	Suivi Probabiliste à Représentation Non Paramétrique . . . . .	12
1.7	Capteur(s) Utilisé(s) . . . . .	13
1.7.1	Suivi Visuel . . . . .	13
1.7.2	Suivi mono-vue, multi-vues . . . . .	13
1.8	Modèle d’Observation . . . . .	14
1.8.1	Modèle d’apparence d’objet . . . . .	14
1.8.2	Modèle d’apparence de fond . . . . .	15
1.8.3	Perturbation de l’Observation . . . . .	15
1.8.4	Modèle d’Observation Individuel, Modèle d’Observation Global . . . . .	19
1.9	Mobilité et paramétrage des objets . . . . .	20
1.10	Suivi Disjoint, Suivi Joint . . . . .	20
1.11	Conclusion : positionnement de la thèse . . . . .	22
<b>2</b>	<b>Méthodes d’Échantillonnage pour Filtres Particulaires</b>	<b>25</b>
2.1	Introduction . . . . .	26
2.2	Échantillonnage de Lois de Probabilité Stationnaires . . . . .	26
2.2.1	Approximation d’une Loi de Probabilité par Échantillonnage . . . . .	26
2.2.2	Choix des Échantillons . . . . .	27
2.3	Échantillonnage de Lois Dynamiques : Filtres Particulaires . . . . .	27
2.3.1	Filtres Particulaires : typologies . . . . .	29
2.3.2	Filtres Particulaires <i>SIR</i> . . . . .	29
2.3.3	Rééchantillonnage par Importance Séquentiel Partitionné . . . . .	30
2.4	Échantillonneurs <i>MCMC</i> . . . . .	34
2.4.1	Échantillonneur de Metropolis . . . . .	36
2.4.2	Échantillonneur de Metropolis-Hastings <i>MH<sub>D</sub></i> . . . . .	37
2.4.3	Échantillonneur de Metropolis-Hastings à propositions marginalisées <i>MH<sub>d</sub></i> . . . . .	37
2.5	Filtres Particulaires par <i>MCMC</i> . . . . .	41

2.5.1	Filtre Particulaire $MCMC_D$	41
2.5.2	Filtre Particulaire $MCMC_d$	42
2.6	Filtre Particulaire $MCMC$ Parallélisé	46
2.7	Conclusion	49
<b>3</b>	<b>Banc d'Essais pour Filtres Particulaires</b>	<b>51</b>
3.1	Introduction	52
3.2	Méthode d'évaluation des performances des filtres	52
3.2.1	Filtres Particulaires	52
3.2.2	Filtre de Kalman	52
3.2.3	Filtre Particulaire contre Filtre de Kalman	53
3.3	Protocole d'expérimentation	54
3.3.1	Nombre de particules, nombre d'évaluations	54
3.3.2	Distance entre lois de probabilité	56
3.3.3	Divergence de Kullback-Leibler	56
3.3.4	Efficacité du rééchantillonnage	57
3.3.5	Critères de performances des Filtres Particulaires	58
3.4	Résultats	58
3.4.1	Effet du nombre d'évaluations	58
3.4.2	Variation de la dimension de l'espace à nombre d'évaluations constant	59
3.4.3	Effet dimension à nombre de particules constant	61
3.4.4	Effet de la dimension du sous espace de propositions de particules	61
3.4.5	Composantes principales de la fonction de vraisemblance	62
3.4.6	Filtre Particulaire $MCMC$ Parallélisé	63
3.5	Conclusion	64
<b>4</b>	<b>Suivi et Catégorisation d'Objets par Filtre Particulaire <math>MCMC</math></b>	<b>67</b>
4.1	Introduction	68
4.2	Suivi d'un nombre fixe d'objets par $FP\ MCMC$	69
4.2.1	Modèle d'état de la configuration jointe multi-objets	69
4.2.2	Modèle d'évolution dynamique des objets	70
4.2.3	Mise en œuvre du $FP\ MCMC$	70
4.2.4	Estimation de l'état de la configuration jointe multi-objets.	71
4.3	Suivi d'un Nombre Variable d'Objets par $FP\ RJ-MCMC$	71
4.3.1	Mouvements proposés	73
4.3.2	Taux d'acceptation	73
4.3.3	Gestion aveugle des mouvements	76
4.3.4	Nécessité de guider les mouvements	76
4.3.5	Mise en œuvre du $FP\ RJ-MCMC$	77
4.4	Modèle d'observation	77
4.4.1	Image masque de particule	79
4.4.2	Segmentation <i>fond / forme</i>	79
4.4.3	Vraisemblance d'observation naïve	79
4.4.4	Vraisemblance relative à la distance objet-caméra	81
4.4.5	Optimisation du calcul de la vraisemblance	82
4.5	Mouvements d'entrée guidés par l'observation	83
4.5.1	Propositions d'entrée guidées en mono-vision	84
4.5.2	Propositions d'entrée guidées en multi-vision	85
4.6	Guidage des mouvements de sortie	85
4.6.1	Mouvements de sortie guidés par l'observation courante	85
4.6.2	Contribution unique des pixels	87
4.6.3	Vitalité des objets présents.	87



4.6.4	Propositions de sortie d'un objet guidées par sa <i>vitalité</i> .	88
4.7	Mouvements guidés : mise en œuvre	88
4.8	Potentiel d'interaction	90
4.8.1	Modélisation des interactions dans la loi dynamique	90
4.8.2	Poids d'interaction	90
4.9	Extension de la méthode au suivi de véhicules	91
4.9.1	Extension du vecteur d'état	91
4.9.2	Modèle dynamique des véhicules	92
4.9.3	Loi de proposition de commande d'un véhicule	92
4.10	Catégorisation des objets intégrée au suivi	92
4.10.1	Intégration de la catégorie dans le vecteur d'état	93
4.10.2	Mise à jour conjointe objet et catégorie	93
4.11	Modélisation et suivi de l'illumination	95
4.11.1	Intégration de l'éclairement dans le vecteur d'état	96
4.11.2	Mouvements additionnels pour l'éclairement	96
4.11.3	Taux d'acceptation d'un mouvement d'éclairement	99
4.11.4	Mise à jour de la <i>vitalité</i> du soleil.	99
4.12	Conclusion	100
<b>5</b>	<b>Expérimentations et résultats</b>	<b>101</b>
5.1	Introduction	102
5.2	Matériau Vidéo et Méthodologie	102
5.3	Cohérence Temporelle des Identités	105
5.3.1	Suivi 2D sur données de synthèse	106
5.3.2	Suivi 3D de piétons en mono-vision sur espace réduit	107
5.3.3	Suivi 3D de piétons en multi-vues	110
5.4	Suivi de moto seule avec vérité GPS	110
5.5	Suivi de piétons sous illumination variable	114
5.6	Suivi et catégorisation de véhicules sous illumination choisie.	116
5.7	Suivi et catégorisation de véhicules sous illumination variable	119
5.8	Suivi et catégorisation de véhicules sur différentes situations	122
5.8.1	Suivi et catégorisation de véhicules sur l'entrée d'une barrière de péage	124
5.8.2	Suivi et catégorisation de véhicules sur route sinueuse	124
5.8.3	Suivi et catégorisation de véhicules sur carrefour giratoire	124
5.9	Conclusion	124
<b>6</b>	<b>Conclusion et Perspectives</b>	<b>129</b>
<b>A</b>	<b>Suivi d'objets joint contre Suivi d'objets disjoint</b>	<b>133</b>
A.1	Evaluation quantitative de la performance du suivi	133
A.2	Résultats	134
<b>B</b>	<b>Échantillonnage Préférentiel</b>	<b>137</b>
B.1	Échantillonnage Préférentiel (ou <i>Importance Sampling, IS</i> )	137
B.2	Échantillonnage Préférentiel Séquentiel (ou <i>Sequential Importance Sampling, SIS</i> )	139
B.3	Rééchantillonnage Préférentiel Séquentiel (ou <i>Sequential Importance Resampling, SIR</i> )	139
<b>C</b>	<b>Performances des Filtres Particulaires : Compléments</b>	<b>143</b>

<b>D</b>	<b>Notations et Conventions</b>	<b>149</b>
D.1	Acronymes . . . . .	150
D.2	Notations . . . . .	151
	<b>Bibliographie</b>	<b>151</b>

# Remerciements

Je tiens à remercier les membres du personnel du LASMEA, qui m'ont accueilli comme un des leurs et m'ont permis de mener à bien le travail présenté dans cette thèse. En particulier, la disponibilité, la compétence et la constante bonne humeur de mon encadrant Thierry Chateau ont été pour moi un puissant moteur. Le portage et l'optimisation en C++ du prototype MATLAB, a été réalisé avec efficacité, rigueur et clairvoyance par Datta Ramadasan lors d'une fructueuse collaboration. Pour ces optimisations, l'usage de la bibliothèque  $NT^2$  <sup>1</sup> a été précieux et mené en collaboration avec Jean-Thierry Lapresté. Je les remercie tous deux chaleureusement. Je remercie mon Directeur de Thèse Michel Naranjo ainsi que le personnel du Rectorat de l'Académie de Clermont-Ferrand, d'avoir œuvré pour que me soient accordés un an de congé de formation puis une année de disponibilité, sans lesquels nos avancées auraient été beaucoup plus modestes. L'année de disponibilité a été financée dans le cadre du projet ANR DIVAS <sup>2</sup>. Je vous remercie ma chère Fabienne et nos chers enfants Adrien, Louis et Marie pour votre infinie patience. Ma chère Maman et mes chers sœurs et frère, je vous remercie pour m'avoir toujours encouragé dans ce travail. Enfin mon cher Papa, tu m'as donné le goût de l'effort et celui d'apprendre, tu tenais à être présent à la soutenance. Tu seras avec nous, et ce travail te doit beaucoup.

---

<sup>1</sup>Numerical Template Toolbox. <http://nt2.sourceforge.net>

<sup>2</sup>Dialogue Infrastructure - Véhicules pour Améliorer la Sécurité routière. <http://or.lcpc.fr/divas-fr/>



# Résumé

Cette thèse présente une méthode de suivi et de classification conjoints en temps réel d'un nombre variable d'objets tels que des piétons et / ou des véhicules, sous conditions d'illumination variables au cours du temps. Le système a été développé au LASMEA<sup>3</sup> de 2006 à 2009, sous la direction de Michel Naranjo et l'encadrement de Thierry Chateau. Nous posons d'abord un panorama des applications nécessitant le suivi et la catégorisation d'objets en temps réel, les problématiques liées au suivi en temps réel de multiples objets, et les méthodes couramment employées. La méthode que nous avons retenue entre dans le champ du suivi Multi-Objets par Filtre Particulaire, dont la clé de voûte est l'échantillonnage des particules. Après avoir rappelé le principe des Filtres Particulaires, nous nous focalisons sur la question du rééchantillonnage des particules. Nous examinons deux familles de filtres particuliers permettant d'opérer dans un espace de grande dimension : les Filtres Particulaires Partitionnés, et les Filtres Particulaires par Chaîne de Markov (*FP MCMC*). Nous comparons ensuite leurs performances sur des données de synthèse. Les résultats obtenus montrent la supériorité du Filtre Particulaire MCMC (méthode de Monte-Carlo par Chaîne de Markov) à mouvements marginalisés. Un système de suivi et classification conjoints en temps réel d'un nombre variable d'objets tels que des piétons et/ou des véhicules, sous illumination variable, est ensuite présenté. La mesure est délivrée par une ou plusieurs caméras statiques. Nous avons délibérément choisi d'alimenter le filtre avec une observation pauvre, reposant uniquement sur une segmentation binaire *avant-plan / arrière-plan* basée sur un modèle de l'*arrière-plan* mis à jour en ligne à chaque image. Pour résister aux variations d'illumination, les ombres sont modélisées et le filtre est étendu afin de suivre conjointement le soleil et les objets au moyen du (*FP MCMC*). Les résultats de suivi et classification en temps réel sont présentés et discutés sur des séquences réelles et sur des séquences de synthèse, impliquant plusieurs catégories d'utilisateurs tels que des piétons, des voitures, des camionnettes et des poids lourds.

**Mots Clef** Suivi Multi-Objets, Suivi Visuel, Temps Réel, Filtre Particulaire MCMC.

---

<sup>3</sup>Laboratoire des Sciences et Matériaux pour l'Électronique et l'Automatique, UMR CNRS - Université Blaise-Pascal



# Abstract

This thesis addresses real-time automatic visual tracking and classification of a variable number of objects such as pedestrians or/and vehicles, under time-varying illumination conditions. The tracking system has been designed in LASMEA<sup>4</sup> Laboratory during years 2006 to 2009, with Michel Naranjo as thesis director and Thierry Chateau as thesis manager. We first draw a landscape of applications where real-time multi-object tracking and classification is required. We then describe the main difficulties related to this functionality, and the most frequently used methods. We choose to use a Particle Filter, operating in the joint multi-object joint configuration. This yields a high dimension state space. Getting efficient operation from a Particle Filter in such a state space is conditioned by the filter ability to « smartly » choose and propagate its particles, a process known as *resampling*. After reminding the principle of Particle Filter, we focus on major strategies for resampling particles : Partitionned Particle Filters and Markov Chain Monte-Carlo Particle Filters (*MCMC PF*). These strategies are benchmarked over synthetic data. Experiments show that we obtain best results with marginal move Markov Chain Monte-Carlo Particle Filters (*MCMC PF*). Based on this choice, a system is described for jointly tracking and classifying a variable number of objects under variable illumination. The measurement is provided by a static camera, and we deliberately chose to feed the filter with poor observation : a basic binary *foreground / background* segmentation. In order to comply with variable illumination, object shadows are modelled and the light source is jointly tracked with the multi-object configuration by the (*MCMC PF*). As a first contribution, we propose in this paper to jointly track the light source within the Particle Filter, considering it as an additionnal object. Illumination-dependant shadows cast by objects are modeled and treated as foreground, thus avoiding the difficult task of shadow segmentation. As a second contribution, we estimate object category as a random variable also tracked within the Particle Filter, thus unifying object tracking and classification into a single process. Real time tracking results are shown and discussed on sequences involving various categories of users such as pedestrians, cars, light trucks and heavy trucks.

**Keywords** Multi-Objet Tracking, Visual Tracking, Real Time, MCMC Particle Filter.

---

<sup>4</sup>Laboratoire des Sciences et Matériaux pour l'Électronique et l'Automatique, UMR CNRS - Université Blaise-Pascal





---

## SUIVI D'OBJETS : PANORAMA

---

Ce chapitre présente d'abord de façon succincte les objectifs fonctionnels visés par notre système de suivi de multiples objets. Nous dressons ensuite un panorama des applications nécessitant le suivi et la catégorisation d'objets en temps réel, ainsi que les contraintes techniques associées.

Nous exposons ensuite les typologies qui se dégagent parmi les techniques les plus fréquemment employées dans le domaine du suivi d'objets visuel. Nous situons enfin nos travaux parmi ces typologies, justifions nos choix méthodologiques et techniques, et précisons les contributions propres à nos travaux.

---

## 1.1 Vue d'ensemble

**Objectifs.** Ce mémoire présente un système de suivi visuel multi-objets développé au LASMEA<sup>1</sup> de 2006 à 2009. Ce système a été développé pour répondre aux besoins d'applications nécessitant le suivi en temps réel d'une scène dynamique complexe, impliquant un nombre élevé d'objets. Le champ d'application principal est la surveillance de lieux publics et d'infrastructures routières. Les objets appartiennent à plusieurs catégories, telles que piétons, cycles, voitures, camionnettes, etc... Plutôt que de développer des réponses dédiées à chaque application, nous avons voulu proposer un système générique standard, qui pourra servir de base à de multiples développements spécifiques à chaque application. Dans cet esprit, le système développé s'appuie sur un minimum de données spécifiques à l'objet : les seuls *a priori* sur l'objet sont un modèle géométrique grossier et un modèle dynamique. Nous n'avons pas utilisé de descripteurs spécifiques basés sur des contenus d'apparence de l'objet de type texture, contours. Nous utilisons également peu de connaissance de la scène : pas de modèle géométrique, pas de zones d'entrée / sortie des objets. Cependant, le système est suffisamment flexible pour prendre en compte ces informations *a priori*, qui ne pourront qu'augmenter ses performances. Un cahier des charges succinct que nous nous sommes fixé est résumé ci dessous :

### Spécifications fonctionnelles

- ▷ suivre jusqu'à une quinzaine d'objets en déplacement sur un sol plan,
- ▷ temps réel vidéo (25 images /seconde),
- ▷ suivi et classification simultanés d'objets de plusieurs catégories,
- ▷ robustesse aux variations d'éclairément,
- ▷ grand ratio surface surveillée / surface au sol des objets suivis : de l'ordre de 2000 en suivi de véhicules ou de piétons,
- ▷ images Noir & Blanc ou couleur basse résolution et fortement compressées de type webcam,
- ▷ acquisition : mono-vue ou multi-vue en fonction de la disponibilité,
- ▷ caméra(s) calibrée(s),
- ▷ caméra(s) statique(s),
- ▷ calculs menés sur un PC grand public,

**Suivi d'objets : définition.** Le suivi d'objets consiste à estimer les positions d'un ou plusieurs objets au cours du temps, à partir de mesures émises par un ou plusieurs capteurs. Cette fonctionnalité est requise par de nombreuses applications. Nous appellerons « *Moteur de Suivi* », ou en bref « *traqueur* », le cœur du système chargé de cette tâche. En fonction du besoin, il peut être demandé au système de suivi, de prendre en charge l'estimation d'informations additionnelles telles que leurs poses, formes, vitesses. Le suivi d'objets étant un processus temporel, la connaissance *a priori* de la dynamique des objets est un moyen d'améliorer la performance du suivi. Si on veut bénéficier de cette connaissance, le suivi ne peut être vu comme une succession de détections d'objets menées indépendamment à chaque image. La détection, en effet, est un processus non temporel consistant à estimer la présence et la position des objets à un instant  $t$ , à partir de la mesure reçue à cet instant. Ce rapport au temps sera détaillé en section 1.5.

---

<sup>1</sup>Laboratoire des Sciences et Matériaux pour l'Électronique et l'Automatique, UMR CNRS - Université Blaise-Pascal

**Applications** Le suivi en temps réel d'un nombre variable d'objets est d'un grand intérêt pour de nombreuses applications telles que la surveillance et l'analyse comportementale de personnes, d'animaux ou de véhicules. Dans le domaine des transports en commun par exemple, le suivi de personnes peut être requis pour analyser des flux de circulation, pour analyser les interactions des usagers avec l'infrastructure, ou simplement du comptage. Dans le domaine de la surveillance, le suivi de personnes peut être utilisé pour détecter des comportements inhabituels ou dangereux, soit en fonctionnement autonome, soit comme pré-traitement délivrant au personnel de surveillance une alarme lorsqu'une telle situation est détectée. L'analyse comportementale des clients dans les centres commerciaux nécessite aussi le suivi de personnes. Dans le domaine des sports, notamment collectifs, des systèmes d'assistance à l'arbitrage sont utilisés. Ces systèmes délivrent aussi des données statistiques individuelles sur chaque joueur. De tels systèmes sont déjà commercialisés, mais ce champ de recherche reste actif car leurs performances restent perfectibles (BvHH09). Les environnements interactifs que l'on développe, pour les musées par exemple, nécessitent souvent de suivre le cheminement des visiteurs. Une étude bibliographique a été publiée (ZMR<sup>+</sup>08). Dans le domaine des *Véhicules Intelligents*, les *Systèmes d'Assistance au Conducteur* (ou *ADAS : Advanced Driver Assistance Systems*), sont des systèmes embarqués qui proposent des fonctionnalités de localisation et/ou de détection d'obstacle en temps réel. De nombreuses applications *ADAS* embarquées s'appuient sur la fusion vision + télémétrie, telles que (RSW08). Certaines applications s'appuient sur la fusion de capteurs proprioceptifs et exteroceptifs tels que l'odométrie, le GPS, le LIDAR, et la vision, qui peuvent être complétées par une cartographie de la route, comme proposé dans (LBCT03). Une telle richesse d'information est rendue nécessaire par la complexité de l'environnement. Elle impose aussi des contraintes, tous ces capteurs devant être calibrés. Dans le domaine des *Systèmes de Transport Intelligents*, le suivi des véhicules est utilisé pour mesurer les flux de véhicules aux intersections, dans le but d'améliorer l'efficacité des transports urbains et routiers (BRB<sup>+</sup>08). La surveillance du trafic routier est également demandée par les applications d'assistance à la conduite et d'anticollision (KCF06).

**Suivi Mono-Objet, Suivi Multi-Objets** Les nombreuses applications ci-dessus demandent parfois le suivi d'un seul objet, mais le plus souvent, elles exigent soit le suivi simultané de plusieurs objets, soit le suivi d'un objet particulier au milieu d'autres objets similaires. Le suivi d'un objet unique est désormais mûr, ayant fait l'objet de travaux depuis de nombreuses années. Dans les applications transport par exemple, le suivi visuel monoculaire d'un véhicule seul a été réalisé en 1993 (KDN93). La voie du suivi visuel par filtrage particulière a été ouverte par Isard et Blake (MA98). Dans cette publication, les auteurs appliquent avec succès leur algorithme *CONDENSATION* au suivi d'un objet unique, tout en insistant sur la multi-modalité de l'observation, causée par un fond encombré ou/et par de multiples cibles.

Une fois que l'on dispose d'une méthode pour suivre un objet unique, on se pose la question de son extension au suivi multi-objets. Le passage d'un à plusieurs objets cibles ne serait pas une difficulté si on arrivait à segmenter les observations de chaque objet cible : il suffirait de lancer un moteur de suivi sur chaque cible. Malheureusement, cette condition n'est généralement pas remplie, notamment dans les applications de surveillance, où les contraintes du lieu limitent l'altitude du capteur. En surveillance de lieux publics intérieurs tels que le métro, l'altitude du capteur est limitée par la hauteur sous plafond ; en surveillance du trafic routier, elle est limitée à l'infrastructure disponible. Ceci génère de fortes occultations entre cibles. Dans ce cas, la segmentation de l'observation selon chaque cible devient difficile voire impossible, et les occultations entre cibles doivent être modélisées. Ce problème est approfondi en section 1.10. Isard *et al.* ont proposé une méthode de suivi multi-objets intégrant une gestion explicite des occultations (IM01), qui a servi de base à de nombreux travaux. Dans le domaine de la surveillance routière, ce problème récurrent est entre autres adressé par (KB05).

Une difficulté supplémentaire est la variabilité temporelle du nombre d'objets suivis (*i.e.* ceux qui sont situés dans le champ de mesure à l'instant  $t$ ) varie au cours du temps. Il faut alors en plus gérer explicitement les entrées et sorties des objets du champ de mesure, comme proposé par exemple par (KBD05). Cette contrainte d'entrées-sorties doit de plus être compatible avec les possibles occultations entre objets. Le modèle d'observation doit donc être soigneusement choisi pour répondre à ces deux exigences, comme cela a été proposé dans (IM01).

**Suivi d'objets de multiples catégories.** Dans les scènes réelles de trafic routier cohabitent différentes catégories d'usagers, telles que camions, camionnettes, voitures, motocycles, cycles et piétons. Selon le capteur utilisé, chaque catégorie d'objet va renvoyer une « signature » de grande variabilité inter-catégorie. Le système doit donc être capable de modéliser cette variabilité, afin d'estimer correctement les positions de cibles de tailles très différentes. Pour réaliser le suivi de voitures dans le trafic réel, la modélisation de la signature d'une voiture est insuffisante. En effet, que le capteur soit optique ou télémétrique, un poids lourd peut très bien présenter une signature qui sera interprétée par le système de suivi comme celle de 2 ou 3 voitures. Une première réponse à ce problème consiste à suivre les cibles avec un unique modèle de signature d'objet générique. Une forte dynamique doit alors être appliquée aux formes des objets candidats, afin de les adapter aux formes des objets cibles. Cette approche est inopérante lorsque les cibles ont des tailles très différentes (cas du trafic routier : rapport de 1 à 5 entre voiture et poids lourd), car rien n'empêche plusieurs objets candidats de petite taille d'expliquer un objet cible de grande taille. Une stratégie en deux étapes - suivi puis catégorisation - est donc inadaptée. Une des contributions majeures du travail présenté dans ce mémoire, est l'intégration de la catégorisation au sein du filtre. Pour ce faire, nous la traitons en variable aléatoire comme les autres composantes de l'état de l'objet. Cette approche présente en outre la possibilité de générer des prédictions dynamiques spécifiques à chaque catégorie d'objet. Le suivi d'un objet par Filtre Particulaire intégrant plusieurs dynamiques a été proposé par (IB98), et appliqué au suivi d'un objet obéissant à des dynamiques très hétérogènes (rebond d'une balle), ou à la catégorisation d'actions d'un même objet (une main au repos, dessinant, ou coloriant). Nous l'appliquons ici à la catégorisation des objets.

## 1.2 Typologies des systèmes de suivi d'objets

Un système de suivi d'objets a pour fonction d'estimer une séquence d'états notée  $\mathbf{X} = \{\mathbf{X}_t\}_{t=1:T}$ . Les lois dynamiques imposent des restrictions aux changements de configuration accessibles par un objet matériel entre deux instants successifs. La séquence d'états peut donc être vue comme une chaîne de Markov d'ordre 1, où l'état  $\mathbf{X}_t$  dépend uniquement de l'état précédent  $\mathbf{X}_{t-1}$ , ainsi qu'illustré par la figure 1.4. Chaque état est dit caché car non directement mesurable, et ne peut être estimé que par le biais d'observations notées  $\mathbf{Z}$ , générées par l'état  $\mathbf{X}$  à travers une transformation directe notée  $f$ , comme illustré sur la figure 1.3. Dans le cas du suivi visuel, les observations sont extraites des images de la scène captées par une ou plusieurs cameras. Un tel système fait donc intervenir au moins trois entités : un « moteur » de suivi, un modèle d'état et un modèle d'observation. Le moteur de suivi met en relation le modèle d'état avec le modèle d'observation, comme schématisé sur la figure 1.1. Si de plus l'état à estimer présente une évolution dynamique modélisable, comme c'est le cas par exemple pour un véhicule, on obtient de meilleures performances en utilisant ce modèle lors de l'étape de prédiction, car on restreint ainsi la région de l'espace d'état à échantillonner, à celle que l'objet peut physiquement atteindre. Le modèle d'évolution constitue alors une quatrième entité à ajouter au schéma de la figure 1.1. Dans les sections suivantes, nous rappelons brièvement les typologies majeures des systèmes de suivi. Une première série de typologies s'établit sur le moteur de suivi :

- ▷ stratégie d'inférence de l'état (section 1.3)
- ▷ suivi probabiliste ou déterministe (section 1.4).
- ▷ rapport au temps (section 1.5).
- ▷ représentation des densités de probabilité en suivi probabiliste (section 1.6).

Une deuxième classification concerne le modèle d'observation :

- ▷ capteurs utilisés (section 1.7)
- ▷ nature du modèle d'observation (section 1.8).

Enfin, le modèle d'état est l'objet de la troisième typologie :

- ▷ suivi 2D contre suivi 3D (section 1.9).
- ▷ Suivi multi-objets disjoint contre suivi joint (section 1.10).

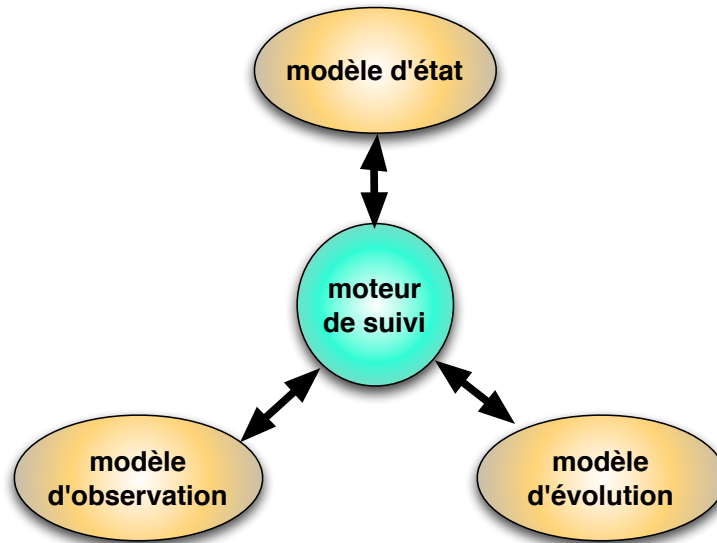


FIGURE 1.1 – Système de suivi d'objets

### 1.3 Méthodes d'inférence d'état pour le suivi d'objets

En suivi d'objets, comme en détection ou en classification, l'opération consiste à estimer l'état  $\mathbf{X}$ , décrivant la configuration des objets dans le monde réel 3D, à partir de l'observation  $\mathbf{Z}$ , délivrée par un ou plusieurs capteurs. L'état est une variable cachée que l'on cherche à inférer à partir d'observations que cet état contribue à générer. La figure 1.3 illustre cette modélisation. Deux stratégies permettent d'inférer l'état caché  $\mathbf{X}$  à partir de l'observation  $\mathbf{Z}$  : 1) l'approche inverse, 2) l'approche directe.

**Méthodes d'inférence inverses.** Ces stratégies consistent à reconstruire l'état d'une scène réelle  $\mathbf{X}$  à partir de l'observation  $\mathbf{Z}$ , en appliquant la transformation  $\mathbf{X} = g(\mathbf{Z})$ , comme indiqué sur la figure 1.3. Elles fournissent alors directement une estimation de l'état à partir des observations, sans passer par des prédictions. Cela sous-entend que l'on soit capable d'établir la transformation  $g$ . Ces approches sont couramment utilisées pour la détection d'objet, où l'état  $\mathbf{X}$  se réduit à une variable binaire définissant la présence ou l'absence de l'objet recherché. Elles sont également utilisées pour le comptage d'objets, où l'état  $\mathbf{X}$  est une variable entière définissant le nombre d'objets présents dans le champ de perception.

**Méthodes d'inférence inverses basées apprentissage.** Une première famille de méthodes permettant d'établir la transformation inverse  $\mathbf{X} = g(\mathbf{Z})$  s'appuient sur un apprentissage hors ligne, utilisant une base d'exemples créée à l'avance, qui contient un ensemble de couples (*observation, état*). Le principal avantage de ces techniques est qu'elles ne nécessitent ni la modélisation de l'état, ni la modélisation de la fonction générant une observation à partir de cet état. Leur inconvénient est que la base d'apprentissage sur laquelle elles s'appuient doit être un sous-ensemble représentatif des états à estimer, ce qui nécessite des bases d'apprentissage énormes dans le cas d'espace d'états de dimension élevée. D'autre part, ces méthodes doivent posséder de bonnes propriétés de généralisation. Parmi les techniques utilisant des bases d'apprentissage, on distingue deux grandes classes de méthodes : les méthodes non paramétriques basées sur une comparaison aux exemples de la base, dans lesquelles la base d'exemples est explicitement stockée en mémoire et sert ensuite de référence pour comparer les nouveaux exemples, et les méthodes basées sur un apprentissage, dans lesquelles un

entraînement, effectué hors ligne, permet de générer un modèle paramétrique qui généralise les propriétés de la base d'exemples. Viola et Jones ont proposé un détecteur d'objet rapide et robuste qui utilise la technique du *Boosting* (VJ01). Dans le cas du suivi d'objets susceptibles de se déplacer dans l'espace, avec des variations de pose, de forme, d'éclairement, et d'occultations, ces approches sont inutilisables car la dimension de l'espace d'état est bien trop élevée pour que l'on puisse constituer une base d'apprentissage représentative. Pour surmonter cette difficulté, Grabner et Bischof ont proposé le *Boosting on-line* (GB06), qui mène simultanément et en temps réel le suivi et l'apprentissage mettant à jour la base d'apprentissage de manière incrémentale, à mesure que de nouvelles observations arrivent.

**Méthodes d'inférence inverses basées modèle.** Ces méthodes s'appuient sur une représentation explicite de la transformation  $\mathbf{X} = g(\mathbf{Z})$ , basée sur un modèle géométrique de reconstruction. Une application de suivi d'objets a été présentée par Yang *et al.* dans (YGB03), où la fonction inverse  $g$  est explicitée par la méthode des *silhouettes visuelles*, ou *visual hulls*, définies comme l'intersection des cônes générant les silhouettes observées sur chaque caméra. Dans cette application, 8 caméras sont utilisées pour dresser une carte de probabilité d'occupation du sol, avec laquelle les objets sont localisés sur l'espace de  $4 \times 4$  mètres. Le nombre de caméras utilisé montre que cette approche requiert une observation riche, pour lever les ambiguïtés causées par les occultations. En mono-vision, les effets de projection et de perspective causent de trop grandes ambiguïtés, rendant cette approche peu exploitable.

**Méthode d'inférence directe.** Le processus consistant à construire une observation à partir d'un état est nommé problème direct, également appelé processus génératif, qui peut s'apparenter à de la synthèse. On ne cherche pas ici à expliciter la fonction la transformation inverse  $\mathbf{X} = g(\mathbf{Z})$ , au contraire, on procède par synthèse : on génère des prédictions d'état  $\mathbf{X}$ , à partir desquelles on génère des observations prédites par la transformation  $\mathbf{Z} = f(\mathbf{X})$ . On les compare ensuite à la vraie observation, puis on met à jour l'état. On est alors dans une approche basée modèle, car sa mise en œuvre requiert trois modèles : un modèle de l'objet, un modèle de l'observation qu'il génère, et une fonction de vraisemblance mesurant la concordance entre l'observation vraie l'observation prédite générée par la transformation  $\mathbf{Z} = f(\mathbf{X})$ . La configuration optimale du modèle, c'est-à-dire celle qui maximise cette fonction de vraisemblance est ensuite estimée. Les méthodes d'estimation basées sur un modèle sont souvent assez précises mais aussi coûteuses en temps de calcul : elles requièrent le calcul d'une fonction de coût très complexe (le modèle doit être rendu dans les images et la fonction de coût évaluée pour chaque hypothèse sur l'état).

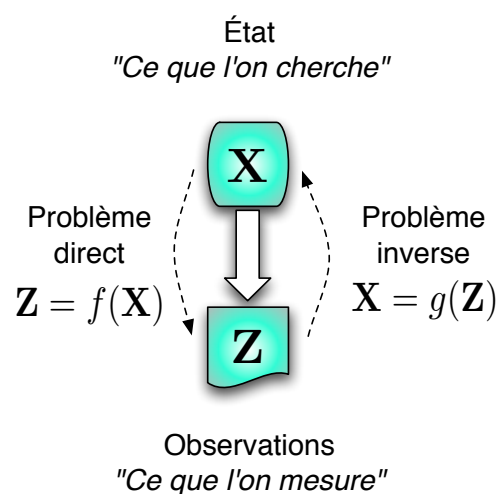


FIGURE 1.2 – Deux approches pour l'estimation de l'état  $\mathbf{X}$ , variable cachée que l'on cherche à inférer à partir de l'observation  $\mathbf{Z}$ . Le processus consistant à construire une observation à partir de l'état est un processus direct qui peut s'apparenter à de la synthèse. L'estimation de l'état à partir d'une observation est un problème inverse, souvent mal posé, et plus complexe à résoudre.

## 1.4 Inférence par résolution directe

On trouve deux grandes classes de méthodes directes : les méthodes probabilistes et les méthodes déterministes. Dans le premier cas, le vecteur d'état est constitué de variables aléatoires, dans le deuxième cas, de variables déterministes. Ces variables peuvent être continues ou discrètes. Les méthodes déterministes s'appuient essentiellement sur des techniques d'optimisation. Les méthodes probabilistes se divisent en deux sous-catégories : les méthodes paramétriques et les méthodes non paramétriques.

### 1.4.1 Inférence d'État Déterministe

Les méthodes déterministes codent le vecteur d'état à l'aide d'un ensemble de variables déterministes. La sortie de l'algorithme associé est alors un ensemble de vecteurs estimant la séquence d'état du système. Un grand nombre de méthodes utilisent des techniques d'optimisation pour estimer l'état du système, en minimisant une fonction de coût. La méthode de suivi et d'estimation de la pose d'un objet 3D en temps réel proposée par Jurie et Dhome (JD01), entre dans cette catégorie : elle consiste à prédire une pose de l'objet, à calculer la différence entre l'image courante et la projection sur le plan image de cette prédiction de la pose de l'objet, puis à utiliser ces différences pour proposer une pose corrigée et à répéter. L'algorithme Meanshift proposé par Comaniciu *et al.* (DVP00), recherche la nouvelle position d'un objet, par une méthode itérative de moyenne glissante. L'état code la position de l'objet dans l'image. Le modèle d'évolution est dit à position constante car la recherche est initialisée autour de la position de l'objet à l'image précédente. Le modèle d'observation est basé sur l'apparence de l'objet dans la première image de la séquence codée à l'aide d'histogrammes couleurs. Le critère optimisé est donc une distance entre histogrammes. L'optimisation peut aussi être menée sur une séquence d'images, comme le proposent Fleuret *et al.* : ils suivent jusqu'à 6 personnes en multi-vision par optimisation globale sur 100 images consécutives (FBLF08).

### 1.4.2 Inférence d'État Probabiliste

L'intérêt d'une méthode probabiliste est qu'elle ne se contente pas d'estimer un état optimum *a posteriori*. Au contraire, elle estime toute la loi de probabilité *a posteriori* de cet état. L'information délivrée est plus riche : dans le cas du suivi, c'est toute cette loi qui est propagée dynamiquement. Dans les méthodes probabilistes, le vecteur d'état  $\mathbf{X}$  et d'observation  $\mathbf{Z}$  sont traités comme des variables aléatoires dont le moteur de suivi doit estimer les distributions de probabilités. La loi de Bayes, équation (1.1) est alors utilisée pour estimer  $p(\mathbf{X}|\mathbf{Z})$ , distribution de probabilité de l'état caché  $\mathbf{X}$ , conditionnée par l'observation  $\mathbf{Z}$ , à partir de  $p(\mathbf{X})$  et  $p(\mathbf{Z})$ , lois de probabilité *a priori* respectives de  $\mathbf{X}$  et  $\mathbf{Z}$ , et de la vraisemblance  $p(\mathbf{Z}|\mathbf{X})$ , représentant le modèle d'observation, c'est à dire la probabilité que l'on observe  $\mathbf{Z}$  lorsque le système est dans l'état  $\mathbf{X}$ .

$$p(\mathbf{X}|\mathbf{Z}) = \frac{p(\mathbf{Z}|\mathbf{X})p(\mathbf{X})}{p(\mathbf{Z})} \quad (1.1)$$

## 1.5 Rapport au temps

Dans le cas d'un système dynamique, l'inférence de son état  $\mathbf{X}$  à partir de l'observation  $\mathbf{Z}$  doit être étendue dans le domaine temporel. Il suffit pour ce faire de définir l'état comme l'ensemble de tous les états successifs entre les instants  $t = 1$  et  $t = T$ , que l'on notera  $\mathbf{X} = \{\mathbf{X}_{1:T}\}$ , et l'observation comme l'ensemble de toutes les observations successivement reçues, noté  $\mathbf{Z} = \{\mathbf{Z}_{1:T}\}$ . Se pose alors la question cruciale de la prise en compte du temps dans la stratégie d'inférence. Les différentes approches envisageables sont décrites dans la suite de cette section. Une première distinction concerne la séquence des états  $\mathbf{X}_{1:T}$  : dans une première famille d'approches, on ne fait aucune hypothèse sur la transition d'un état au suivant. Ceux-ci sont donc considérés comme étant indépendants les uns des autres, comme illustré par la figure 1.4. Dans d'autres approches, la séquence d'états est modélisée par hypothèse comme une chaîne de Markov d'ordre 1, comme illustré par la figure 1.4. L'état  $\mathbf{X}_t$  dépend alors de l'état précédent  $\mathbf{X}_{t-1}$ . Ce choix est lourd de conséquences sur la méthodologie d'inférence.



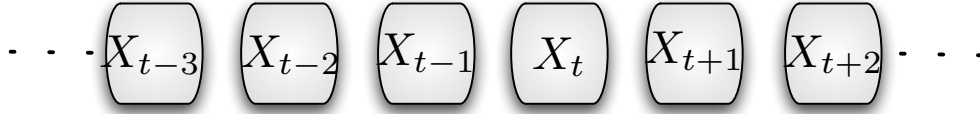


FIGURE 1.3 – Une séquence d'états indépendants. Chacun des états  $\mathbf{X}_t$  est indépendant des autres.

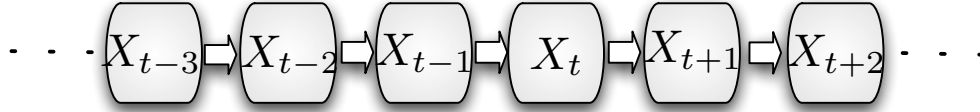


FIGURE 1.4 – Une séquence d'états modélisée comme une chaîne de Markov d'ordre 1. L'état  $\mathbf{X}_t$  dépend uniquement de l'état précédent  $\mathbf{X}_{t-1}$ .

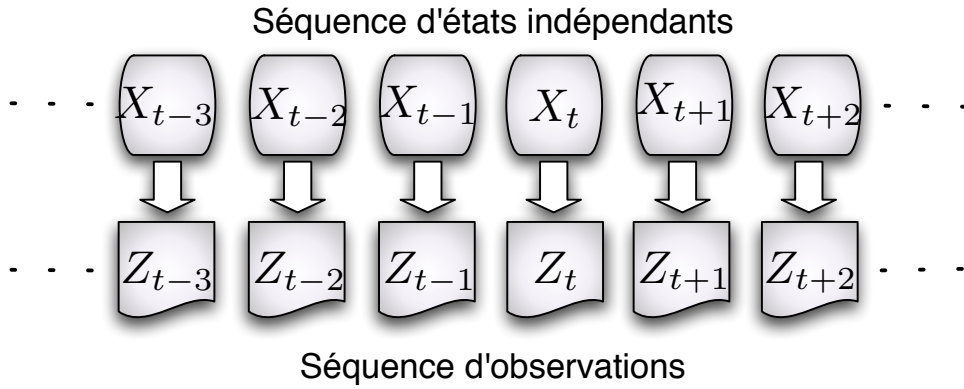


FIGURE 1.5 – Séquence d'états cachés  $\mathbf{X} = \{\mathbf{X}_{1:T}\}$  d'un système dynamique, et les observations  $\mathbf{Z} = \{\mathbf{Z}_{1:T}\}$  qui en résultent.

### 1.5.1 Détection d'état instantané

Une *détection d'état instantané* applique un algorithme de détection à chaque image considérée isolément. Dans cette approche, la tâche du détecteur consiste à inférer l'état  $\mathbf{X}_t$  à chaque instant  $t$ , à partir de la seule observation  $\mathbf{Z}_t$ . Cette estimation n'a pas de caractère temporel, n'est pas produite par un filtre, et ne peut prendre en compte un modèle dynamique de l'état. Le choix de cette approche se justifie notamment lorsqu'on n'a pas de modèle dynamique de l'état. Dans le cas d'une inférence probabiliste, l'équation 1.2 est appliquée. Özuysal *et al.* proposent une méthode d'estimation de la position et de la pose 3D d'un objet dont on n'a pas de modèle géométrique ni dynamique, sur la base d'attributs visuels dans la seule image courante (ÖLFF06) (un modèle de la dynamique de la pose de la caméra est cependant mis en œuvre). Pour du suivi multi-objets, une telle approche est obligatoirement suivie d'une association de données, tâche rendue ardue par le manque de cohérence des résultats de détections d'une image sur l'autre : le détecteur ne délivre à chaque image qu'un ensemble discret de réponses positives, dont de fausses positives, et dont des détections peuvent être absentes (faussement négatives).

$$p(\mathbf{X}_t | \mathbf{Z}_t) = \frac{p(\mathbf{Z}_t | \mathbf{X}_t) p(\mathbf{X}_t)}{p(\mathbf{Z}_t)} \quad (1.2)$$



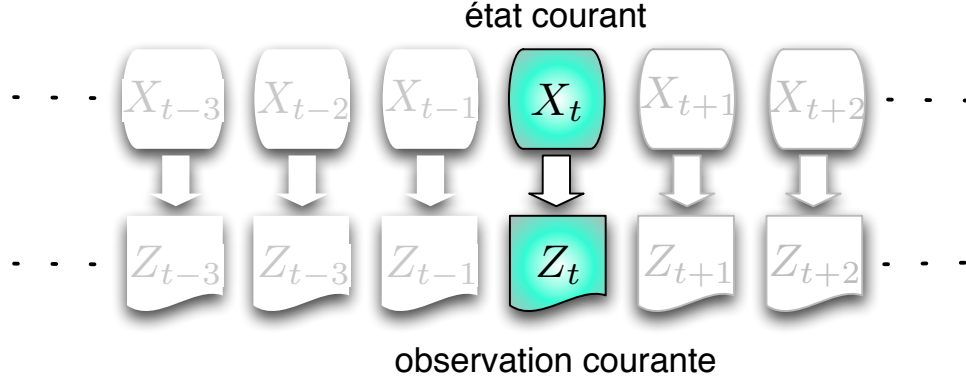


FIGURE 1.6 – Suivi par détection. Seule l’observation courante (colorée en bleu) est prise en compte pour inférer le seul état courant (coloré en bleu).

### 1.5.2 Estimation de trajectoire *a posteriori*

Dans cette approche hors ligne non-séquentielle, on alimente le moteur d’inférence avec la totalité des observations du début à la fin de la séquence, comme illustré par la figure 1.7. La séquence d’états  $\mathbf{X} = \{\mathbf{X}_{1:T}\}$  est vue comme l’union des états successifs, supposés indépendants. Cette approche est sans aucun doute la plus performante puisque la totalité de la séquence d’observations  $\mathbf{Z} = \{\mathbf{Z}_{1:T}\}$  est prise en compte pour l’estimation simultanée de la totalité de la séquence d’états  $\mathbf{X} = \{\mathbf{X}_{1:T}\}$ , c’est à dire de la trajectoire complète. L’application de la loi de Bayes à cette séquence, équation (1.3), permet d’estimer *a posteriori* la globalité de la trajectoire.

$$p(\mathbf{X}_{1:T}|\mathbf{Z}_{1:T}) = \frac{p(\mathbf{Z}_{1:T}|\mathbf{X}_{1:T})p(\mathbf{X}_{1:T})}{p(\mathbf{Z}_{1:T})} \quad (1.3)$$

L’estimation n’a pas de caractère temporel, elle n’est pas produite par un filtre, et nous ne classons pas cette approche comme du suivi. Une telle approche reposant sur un échantillonnage par chaîne de Markov pour explorer l’espace des trajectoires d’objets sur toute une séquence, a été proposée par Medioni *et al.* (YMC07). Cependant, comme dans le cas du suivi, on est ici confronté à un volume d’information d’autant plus imposant que la séquence est longue. Un deuxième problème est la nécessité d’un pré-traitement pour segmenter temporellement la séquence. Enfin, bien entendu, cette approche ne se prête pas aux applications temps réel.

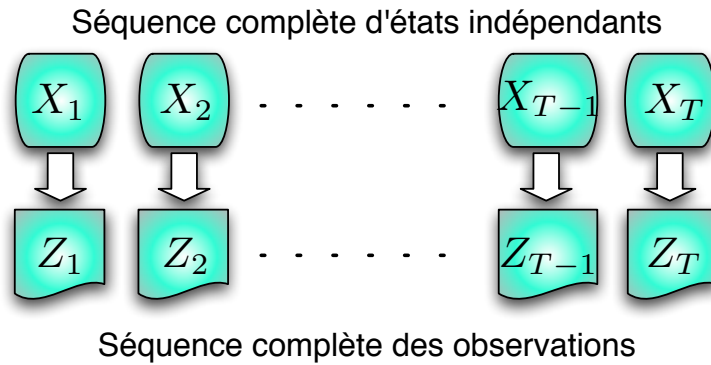


FIGURE 1.7 – Estimation de trajectoire *a posteriori*. L’intégralité de la séquence des observations (colorées en bleu) sont prises en compte pour inférer l’intégralité de la séquence des états (colorés en bleu).

### 1.5.3 Suivi d'État Causal

Les applications temps réel imposent une approche causale, où seules les observations passées sont disponibles. A chaque période, la nouvelle observation sera prise en compte dès qu'elle est disponible, pour alimenter le moteur de suivi. On dispose ainsi de toutes les observations présente et passées, comme illustré par la figure 1.8. Cette richesse est par exemple exploitée par les travaux de Fleuret *et al.*, qui suivent jusqu'à 6 piétons en multi-vision et en caméras fixes, en menant une optimisation spatio-temporelle pour chaque piéton sur une fenêtre temporelle de 100 images (FBLF08). L'optimisation est ainsi menée sur une observation beaucoup plus cohérente. Le grand avantage de cette approche est qu'elle ne repose pas sur l'hypothèse de transition markovienne de l'état  $\mathbf{X}_t$  au suivant  $\mathbf{X}_{t+1}$ , contrairement à la méthode séquentielle décrite en section 1.5.4. Il en résulte une meilleure robustesse du suivi, car une perte de suivi due à une observation temporairement défaillante, sera récupérée dès que l'observation redeviendra bonne. Cet avantage est revendiqué et démontré par Leibe *et al.*, qui étendent cette approche en menant une optimisation spatio-temporelle conjointe sur tous les objets présents (LSCG08). Le couplage de cette méthode avec des détecteurs performants leur permet de plus de mener ce suivi depuis des caméras embarquées (LSS).

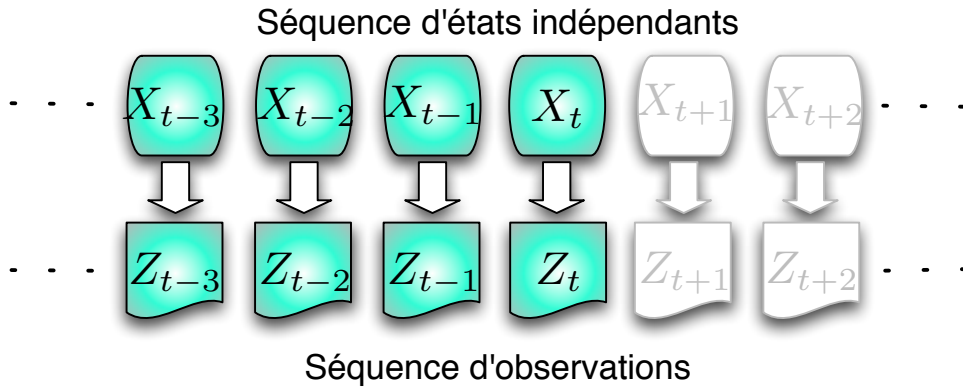


FIGURE 1.8 – Suivi d'état causal. L'intégralité des observations passées (colorées en bleu) sont prises en compte pour inférer l'intégralité des états passés (colorés en bleu).

### 1.5.4 Suivi d'État Causal Séquentiel

Il apparaît sur la figure 1.8, que le volume d'observations disponibles  $\mathbf{Z} = \{\mathbf{Z}_{1:t}\}$  pour le suivi, s'accroît à mesure que le temps passe. Le suivi causal séquentiel permet d'éviter de traiter un tel volume, au prix d'une hypothèse : la séquence d'états se modélise comme une chaîne de Markov conditionnant la transition de l'état  $\mathbf{X}_{t-1}$  à l'état  $\mathbf{X}_t$ . L'inconvénient de cette stratégie est qu'une perte temporaire du suivi est difficilement récupérable, comme l'ont argumenté Leibe *et al.* dans (LSCG08). Une approche probabiliste est alors préférable à une optimisation, afin de minimiser ce risque de défaillance du suivi. Dans le cas d'une approche probabiliste, l'équation (1.1) peut encore s'appliquer pour estimer *a posteriori* la trajectoire composée des  $t$  échantillons temporels passés  $\mathbf{X} = \{\mathbf{X}_{1:t}\}$ , à partir des  $t$  observations passées  $\mathbf{Z} = \{\mathbf{Z}_{1:t}\}$ , selon l'équation (1.4).

$$p(\mathbf{X}_{1:t}|\mathbf{Z}_{1:t}) = \frac{p(\mathbf{Z}_{1:t}|\mathbf{X}_{1:t})p(\mathbf{X}_{1:t})}{p(\mathbf{Z}_{1:t})} \quad (1.4)$$

Comme illustré par la figure 1.9, cette stratégie estime à chaque instant l'état courant  $\mathbf{X}_t$ , à partir de l'état précédent  $\mathbf{X}_{t-1}$  et de l'observation courante  $\mathbf{Z}_t$ . Dans une approche probabiliste, le suivi séquentiel permet d'estimer récursivement la loi de probabilité *a posteriori*  $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$  de l'état  $\mathbf{X}_t$  à l'instant  $t$ , conditionné à l'historique des mesures  $\mathbf{Z}_{1:t}$ . Dans ces conditions, la règle de Bayes devient :

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t}) = \frac{p(\mathbf{Z}_t|\mathbf{X}_t)p(\mathbf{X}_t|\mathbf{Z}_{1:t-1})}{p(\mathbf{Z}_t|\mathbf{Z}_{1:t-1})}, \quad (1.5)$$

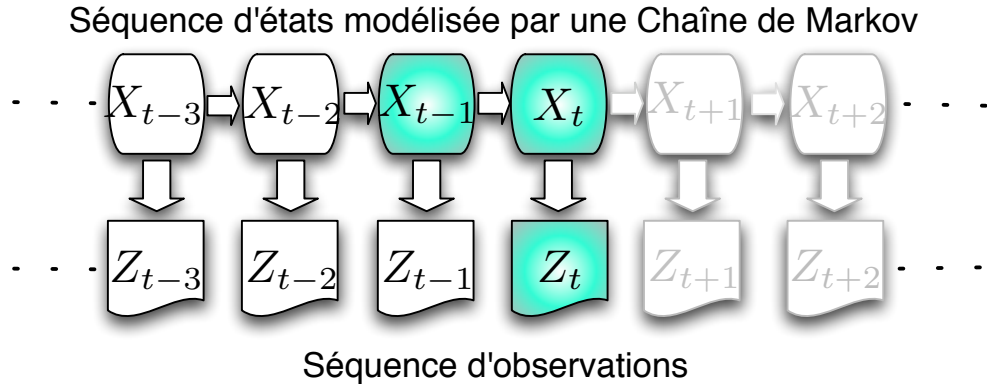


FIGURE 1.9 – Suivi d'état Causal Séquentiel. Seule l'observation courante et l'état précédent (colorés en bleu) sont pris en compte pour inférer le seul état courant (coloré en bleu).

où la loi de probabilité *a priori*  $p(\mathbf{X}_t | \mathbf{Z}_{1:t-1})$  est explicitée par l'équation de Chapman-Kolmogorov (1.6) :

$$p(\mathbf{X}_t | \mathbf{Z}_{1:t-1}) = \int_{\mathbf{X}_{t-1}} p(\mathbf{X}_t | \mathbf{X}_{t-1}) p(\mathbf{X}_{t-1} | \mathbf{Z}_{1:t-1}) d\mathbf{X}_{t-1}. \quad (1.6)$$

Dans l'équation (1.6),  $p(\mathbf{X}_t | \mathbf{X}_{t-1})$  est la loi de probabilité d'évolution temporelle de l'état, qui génère les prédictions pour l'instant  $t$ , et  $p(\mathbf{X}_{t-1} | \mathbf{Z}_{1:t-1})$  est la loi de probabilité *a posteriori* à l'instant  $t-1$ . Le remplacement de l'équation (1.6) dans l'équation (1.1) conduit à l'expression du filtre Bayésien récursif :

$$p(\mathbf{X}_t | \mathbf{Z}_{1:t}) = C^{-1} p(\mathbf{Z}_t | \mathbf{X}_t) \int_{\mathbf{X}_{t-1}} p(\mathbf{X}_t | \mathbf{X}_{t-1}) p(\mathbf{X}_{t-1} | \mathbf{Z}_{1:t-1}) d\mathbf{X}_{t-1}, \quad (1.7)$$

où  $C = p(\mathbf{Z}_t | \mathbf{Z}_{1:t-1})$  est une constante indépendante de l'état. Cette équation permet de réaliser le suivi probabiliste sur des durées illimitées, puisque  $p(\mathbf{X}_t | \mathbf{Z}_{1:t})$ , loi de probabilité *a posteriori* de l'état  $\mathbf{X}$  à l'instant  $t$ , est déduite récursivement du modèle d'observation  $p(\mathbf{Z}_t | \mathbf{X}_t)$ , du modèle dynamique d'état  $p(\mathbf{X}_t | \mathbf{X}_{t-1})$ , et de  $p(\mathbf{X}_{t-1} | \mathbf{Z}_{1:t-1})$ , loi *a posteriori* à l'instant  $t-1$ . Le travail présenté dans les chapitres suivants repose sur une méthode de suivi causal séquentiel.

## 1.6 Représentation des Loïs de Probabilités pour le Suivi Probabiliste

### 1.6.1 Suivi Probabiliste à Représentation Paramétrique

Dans une première catégorie d'approches probabilistes, on fait des hypothèses sur les densités de probabilité (et sur le modèle d'évolution dans le cas d'un suivi séquentiel), que l'on peut alors décrire par un petit nombre de paramètres, ce qui diminue les volumes de calculs. L'hypothèse la plus fréquemment employée consiste à supposer que le modèle d'évolution et le modèle d'observation sont gaussiens, et que le modèle d'observation est linéaire, ce qui autorise l'utilisation d'un filtre de Kalman. Dans ce cas, le filtre délivre une estimation optimale de l'état des objets, comme proposé par exemple dans (Gen92). Malheureusement, en suivi visuel, ces hypothèses ne sont presque jamais vérifiées, ce qui conduit à des défaillances. La cause principale de ces défaillances, illustrée par Isard *et al.* dans (MA98), est due à la présence de maxima locaux dans le modèle d'observation, qui « distraient » le *traqueur*, et le décalent de sa cible, si bien que le *traqueur* peut rester bloqué sur ces maxima, abandonnant sa cible. Isard *et al.* ont montré qu'il suffit, pour que se produise ce scénario, que l'arrière-plan de la scène soit suffisamment encombré pour que le détecteur utilisé (un contour actif dans leur cas), génère des maxima locaux sur l'arrière-plan. La figure 1.10 illustre le modèle d'observation issu d'une segmentation *fond / forme* classiquement utilisée lorsque la caméra est statique. Cette représentation met en évidence des caractéristiques du modèle d'observation global, contradictoires à une modélisation gaussienne :

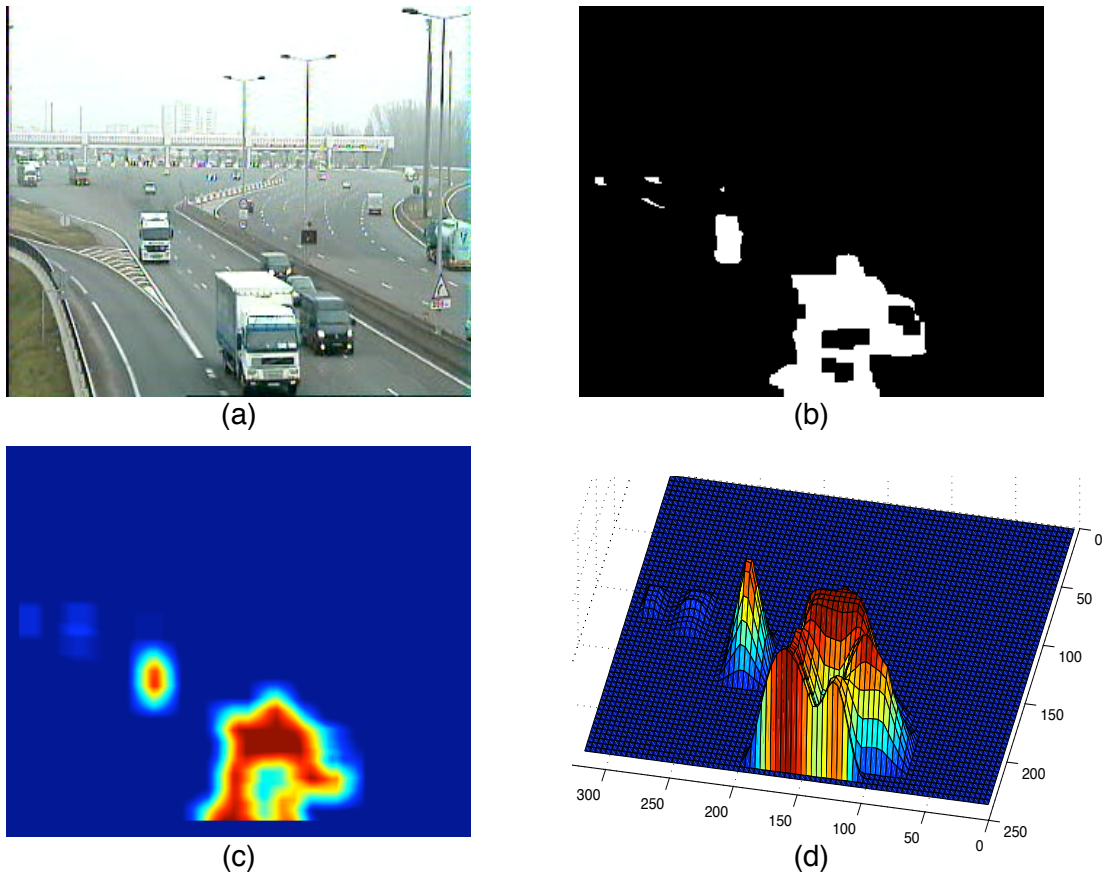


FIGURE 1.10 – Fonction d’observation après segmentation *fond / forme* en caméra statique. (a) : image brute. (b) : image binaire *fond / forme*. (c) et (d) : modèle d’observation obtenu par convolution de l’image (b) par un masque rectangulaire uniforme de taille arbitrairement fixée à  $20 \times 20$  pixels.

- ▷ impossible à partitionner entre les objets, dès qu’ils s’intersectent,
- ▷ éventuelle multi-modalité en cas de forte occultation,
- ▷ observation nulle dès que le noyau n’englobe aucun pixel segmenté comme *forme*.

### 1.6.2 Suivi Probabiliste à Représentation Non Paramétrique

Dans cette approche, on ne fait aucune hypothèse ni sur les densités de probabilité ni sur le modèle d’évolution. Dans ce cas, l’intégrale contenue dans l’équation (1.7) ne peut en général pas être calculée analytiquement. Une solution est alors de recourir à l’échantillonnage pour approximer la loi de probabilité de l’état caché (non observable)  $\mathbf{X}$  d’un système dont le modèle d’évolution et/ou le modèle d’observation  $\mathbf{Z}$  sont quelconques (en particulier non gaussiennes), au moyen de  $\{\mathbf{X}^n, \pi^n\}_{n=1}^N$ , ensemble de  $N$  échantillons discrets  $\mathbf{X}^n$  pondérés des poids associés  $\pi^n$ , tels que :

$$p(\mathbf{X}|\mathbf{Z}) \approx \sum_{n=1}^N \pi^n \delta(\mathbf{X} - \mathbf{X}^n), \quad (1.8)$$

où  $\delta$  est l'impulsion de Dirac, et  $\pi^n$  est le poids du  $n^{\text{ième}}$  échantillon,  $n \in 1 \dots N$ , tel que  $\sum_{n=1}^N \pi^n = 1$ . Cette approche est développée dans le chapitre 2.

## 1.7 Capteur(s) Utilisé(s)

Les capteurs les plus utilisés dans le domaine du suivi multi-objets sont la télémétrie par laser ou par ultrasons, et l'imagerie optique. Les télémètres offrent des étendues de mesures limitées à quelques dizaines de mètres et restent très coûteux. Ils offrent cependant une bonne complémentarité d'observation par rapport à la vision, qui incite à fusionner ces deux modalités de mesure. Cette fusion est très utilisée pour de nombreuses applications, notamment dans les applications liées aux véhicules, car elle permet d'atteindre de bonnes performances, comme exposé dans (GCT09). Nous nous positionnons dans une situation où l'information disponible est plus modeste : une ou plusieurs caméras de faible résolution. Dans ce mémoire, nous nous limiterons donc à l'utilisation de la vision seule.

### 1.7.1 Suivi Visuel

La vision présente les avantages suivants :

- ▷ Grande richesse d'information : les traitements d'images couramment utilisés permettent d'extraire des informations de couleur, de texture, de forme, de points d'intérêts.
- ▷ Grande étendue de mesure : un champ de mesure de 45 degrés d'ouverture et de 200 mètres de profondeur est couramment exploitable en suivi de trafic routier par exemple.
- ▷ Faibles coûts et contraintes d'installation.

D'un autre côté, les difficultés liées à la vision sont les suivantes :

- ▷ Lourdeur des traitements : c'est le revers de la grande richesse d'information !
- ▷ Forte sensibilité aux conditions d'éclairement et aux ombres portées.
- ▷ Forte sensibilité à la pose des objets et aux occultations.
- ▷ Forte sensibilité au bruit dû par exemple à un fond d'image encombré.

### 1.7.2 Suivi mono-vue, multi-vues

Le choix de la mono-vision se justifie par la faiblesse des coûts de matériel et d'installation, ainsi que par la limitation du volume d'information à traiter, ce qui est crucial pour toutes les applications temps réel. Le choix de la mono-vision est parfois imposé par le cahier des charges de certaines applications, où on ne dispose que d'une caméra pour couvrir un espace donné, en vidéo-surveillance par exemple. Pour ces raisons, la majorité des travaux publiés en suivi visuel sont basés sur la mono-vision. Cependant, la mono-vision présente une faiblesse inhérente : une mauvaise observabilité de la profondeur, défaut d'autant plus gênant que la caméra est placée à une faible altitude, comme c'est souvent le cas en vidéo-surveillance. Cette faiblesse a souvent été pointée, et des moyens pour minimiser ses effets ont été proposés (KB05).

Pour surmonter cette difficulté, il peut être avantageux de fusionner l'information provenant de plusieurs caméras, ou de fusionner l'information caméra avec celle provenant d'autres capteurs. Nous nous limiterons ici aux capteurs optiques, il s'agira donc de fusionner les informations provenant de plusieurs points de vue. L'apport de la vision multi-vue est évidemment une meilleure observabilité, comme mis en évidence par Fleuret *et al.* dans (FBLF08). En contrepartie, la vision multi-vue impose des contraintes supplémentaires : les caméras doivent être calibrées, et synchronisées. Des difficultés de traitement supplémentaires apparaissent aussi : la première difficulté est la définition d'une stratégie de fusion. Cette difficulté est accrue lorsque les champs de vision ne se recouvrent que partiellement. La deuxième difficulté est une augmentation du temps de traitement,

du fait du volume d'information supérieur. Ceci est pénalisant pour toutes les applications temps réel. Nous avons conçu un moteur de suivi prenant en compte l'observation émise par un nombre quelconque de caméras statiques calibrées, c'est à dire apte à fonctionner aussi bien en mono-vision, qu'en multi-vues. Nous nous limitons néanmoins au cas du suivi d'objets contenus dans une zone incluse dans l'intersection des champs de vision des caméras. Ce modèle d'observation est défini au chapitre 4. La méthode présentée dans ce manuscrit pourrait aussi autoriser la fusion avec d'autres capteurs.

## 1.8 Modèle d'Observation

Les applications de suivi d'objets peuvent encore être classées en deux familles, selon que la caméra est dynamique ou statique. Le premier cas se présente notamment dans les applications embarquées telles que les Systèmes d'Assistance au Conducteur (ADAS), ou dans les applications de vision pour la robotique embarquée. Pour ces applications, la scène est inconnue *a priori*, et l'observation est obligatoirement basée sur l'apparence des objets à suivre, donc sur un détecteur de ces objets. Cette approche est décrite dans la section 1.8.1. Lorsque la caméra est statique, il est avantageux de modéliser l'apparence de la scène vide, comme décrit dans la section 1.8.2.

### 1.8.1 Modèle d'apparence d'objet

Dans les approches de *Suivi par Détection* ou *Tracking by Detection*, l'observation est basée sur des scores établis par un détecteur. Ce champ de recherche est très actif et rendu nécessaire par les applications mobiles, où la scène est inconnue *a priori* (OTF<sup>+</sup>04; GM07; LCCVG07; WN07). Plusieurs modalités de contenu visuels sont couramment utilisés pour générer le modèle d'observation d'un objet : contenu colorimétrique, forme, texture, contours, points d'intérêt, mouvement... La variété de ces fonctions d'observations constitue un vaste champ que nous ne voulons pas couvrir ici. Nous citons seulement quelques exemples ci-dessous.

**La Couleur** Le contenu colorimétrique d'un objet est évidemment un indice visuel de première importance. Ce contenu constitue de plus un descripteur permettant de différencier des objets de géométrie similaire entre eux, tels que des véhicules ou des personnes. Le choix de l'espace de couleurs dans lequel on décrit le contenu colorimétrique de l'objet, fait l'objet de nombreux travaux. L'espace *RVB* (Rouge Vert Bleu, ou *RGB* : Red Green Blue) est vite évacué car trop sensible aux variations d'illumination. Le choix est alors délicat et dépendant de l'application, entre l'espace *rvb* normalisé, qui est très simple à calculer, ou *HSV* (Hue Saturation Value, ou Teinte Saturation Intensité), qui l'est moins, ou d'autres encore comme *YCrCb* (Intensité, Chrominance rouge, Chrominance bleue)...

La modélisation d'apparence d'un objet par un histogramme de son contenu colorimétrique est une approche très utilisée pour suivre des objets déformables tels que des personnes, car ce contenu est relativement invariant aux variations de posture (DVP00). Une représentation non paramétrique de cette distribution de couleurs a été proposée par Perez *et al.* dans (PCJM02). La représentation se fait dans l'espace couleur *HSV* et consiste en un histogramme *HS+V*, composé d'un histogramme joint de teinte et de saturation, dans lequel n'entrent que les pixels d'intensité suffisante (typiquement  $>0.1$  sur une échelle de 0 à 1), et d'un histogramme d'intensité rempli par les autres pixels. Néanmoins, la représentation du contenu colorimétrique d'un objet par un unique histogramme ne permet pas de prendre en compte la distribution géométrique des couleurs sur l'objet (c'est l'objectif recherché). Mais Perez *et al.* montrent dans le même article, que l'on obtient un meilleur suivi en découpant l'objet en tranches, et en décrivant le contenu de chacune par un histogramme. Ce faisant, on redonne du poids à la distribution géométrique des couleurs dans l'objet. Sur du suivi de personnes, les performances sont alors améliorées avec 3 tranches (jambes, torse, tête). Si cette représentation par histogramme de contenu colorimétrique est robuste aux déformations de l'objet, elle est sensible aux variations d'éclairement, ce qui nécessite une mise à jour périodique du modèle. Elle est également peu robuste aux occultations de l'objet.



**La forme** Elle est également un indice visuel de première importance. Par rapport à la couleur, elle est par contre très sensible aux variations de pose, d'échelle et de perspective. Elle est donc obligatoirement mise à jour périodiquement. Isard et Blake ont démontré les performances de leur algorithme *CONDENSATION* sur du suivi de personnes (tête et épaules), de main, et de feuilles végétales, en utilisant un modèle d'observation basé sur un contour actif : une spline paramétrique (MA98). Les paramètres de cette spline sont portés par le vecteur d'état, et suivis par le filtre avec un modèle d'observation composé des gradients selon les directions normales à la spline en une collection de points. Ce modèle d'observation se calcule à partir d'une image de gradients. Ils montrent qu'elle est très sensible à un fond encombré, qui génère des multi-modalités. Mac Cormick et Blake ont ensuite généralisé cette approche au suivi de multiples objets, en utilisant ce même modèle d'observation, montrant sa robustesse aux occultations partielles (MB99).

### 1.8.2 Modèle d'apparence de fond

Lorsque la caméra est statique, la scène est connue *a priori*, et il est avantageux d'exploiter cette connaissance par un apprentissage de l'apparence du *fond*, c'est à dire de la scène vide, ce qu'exploitent de nombreux travaux, dont ceux de Fleuret *et al.* (FBLF08). C'est une approche locale basée pixel, dite *soustraction de fond* ou *segmentation fond / forme*. Ce traitement de bas niveau présente l'avantage de renvoyer une observation robuste aux variations de pose et aux déformations des objets suivis. Dans tous les cas, il consiste à confronter l'image courante à un modèle du *fond* supposé vide de toute *forme*. Un modèle simple consiste à représenter la valeur de chaque pixel dans l'espace de couleur choisi, par une gaussienne multivariée, et à déclarer qu'un pixel de l'image courante appartient à la *forme*, si sa distance de Mahalanobis au modèle excède un seuil choisi. Cette approche est mise en défaut par les variations d'illuminations. Dans certaines applications (surveillance de lieux publics très fréquentés, autoroutes par exemple), il est de plus impossible d'obtenir une image du *fond* libre de toute *forme*. Dans ces conditions, une adaptation en ligne du modèle est indispensable, comme proposé par (Wre97) par exemple. Cette modélisation unimodale du *fond* montre ses faiblesses dès que le *fond* comporte plusieurs modes (par exemple sur du bougé de caméra). Stauffer et Grimson ont proposé une modélisation beaucoup plus performante : la valeur de chaque pixel est représentée par une mixture adaptative de gaussiennes multivariées dans l'espace de couleur choisi (Sta99). Cette modélisation a connu un grand succès depuis, mais elle est coûteuse en calculs. Une étude bibliographique des développements dans le domaine de la modélisation du *fond* par mixture de gaussiennes a été publiée récemment (BEBV08). Le champ de recherche dans le domaine de la *segmentation fond / forme* est immense et très actif, et plutôt que de nous y engager, nous avons préféré utiliser l'algorithme  $\Sigma - \Delta$  (MR04), qui calcule efficacement en ligne un modèle adaptatif du fond à partir duquel est menée la segmentation.

### 1.8.3 Perturbation de l'Observation

**Ombres portées des objets** Les ombres portées des objets sur le fond peuvent considérablement perturber l'observation, notamment l'apparence des objets et celle du *fond*. Elles perturbent par conséquent la segmentation *fond / forme*. C'est notamment le cas en extérieur, sous soleil direct générant des ombres portées très contrastées. Les figures 1.11 et 1.12 montrent que les ombres générées par de telles conditions d'éclairement sont en très grande partie classifiées comme *forme* par la segmentation *fond / forme*, et que la surface de leur projection à l'image est du même ordre de grandeur que celle des objets eux-mêmes. Cette perturbation de la segmentation diminue la précision de suivi, car les objets estimés se trouvent attirés par les ombres de leurs objets cibles. En outre, elle génère des erreurs de suivi lorsque le système instancie à tort un objet candidat sur l'ombre d'un objet cible. Ces deux défaillances ont été décrites dans la littérature, par (PMT03; SCE04) entre autres. Cependant, les ombres des objets portent aussi des informations pertinentes sur la position et la forme de l'objet, offrant la possibilité d'accroître leur observabilité, ce que l'on constate sur la figure 1.12, où les ombres des véhicules apportent une information discriminante sur leur position longitudinale, ainsi que sur la simple présence du poids lourd mal segmenté dans l'image #10360. Pour ces raisons, les ombres doivent être prises en compte afin d'améliorer les performances du suivi visuel (SCE04). Une étude bibliographique et un comparatif de performances de différents algorithmes de segmentation d'ombre, ont été menés et publiés dans (PMT03). Néanmoins, la segmentation de l'image en trois classes : *fond / forme / ombre de forme*, est très

difficile si on se limite au niveau pixelique, ce qui a conduit les auteurs à ajouter des heuristiques spatiales et temporelles dans leurs méthodes de segmentation. Considérant les difficultés liées à cette segmentation ternaire, nous avons préféré mener une segmentation binaire *fond / forme + son ombre*, associée à une modélisation de la génération des ombres portées des objets mobiles. Nous adoptons une méthode similaire. Leotta *et al.* (MJL06) associent une modélisation 3-D des objets, avec un éclairage défini manuellement par l'utilisateur, pour générer leurs ombres portées au sol. Cette approche est efficace sur un court laps de temps, mais dès que l'éclairage change, il faut redéfinir ses paramètres. Nous étendons cette approche en intégrant les paramètres d'éclairage au vecteur d'état de la scène, et confions leur suivi au moteur de suivi, ce qui permet de suivre les objets sur de longues périodes.

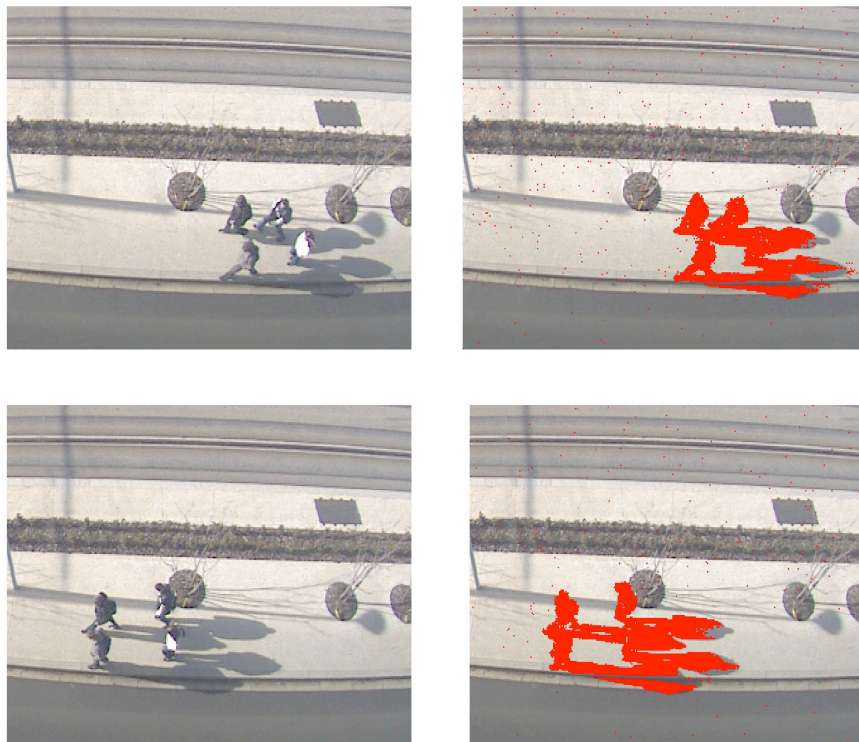


FIGURE 1.11 – Segmentation *fond / forme* dans l'espace couleurs RGB, de piétons sur voie publique. A gauche : images brute. A droite : les pixels classifiés comme *forme* sont coloriés en rouge.

**Reflets des objets** Les reflets des objets sur des surfaces réfléchissantes peuvent elles aussi considérablement perturber la segmentation *fond / forme*. C'est notamment le cas sur les sols en pierre polie, comme illustré figure 1.13 (image issue de la base *caviar*<sup>2</sup>). Cette figure montre que les reflets de la personne sur le sol lisse, ainsi que sur la vitrine gauche, sont en très grande partie classifiés comme *forme* par la segmentation *fond / forme*, et que la surface de leur projection à l'image est du même ordre de grandeur que celle des objets eux-mêmes. De même qu'avec les ombres, cette perturbation de la segmentation diminue la précision de suivi, car les objets estimés se trouvent attirés par les reflets de leurs objets cibles. En outre, elle génère des erreurs de suivi lorsque le système instancie à tort un objet candidat sur le reflet d'un objet cible.

<sup>2</sup>[www.inria.fr](http://www.inria.fr)



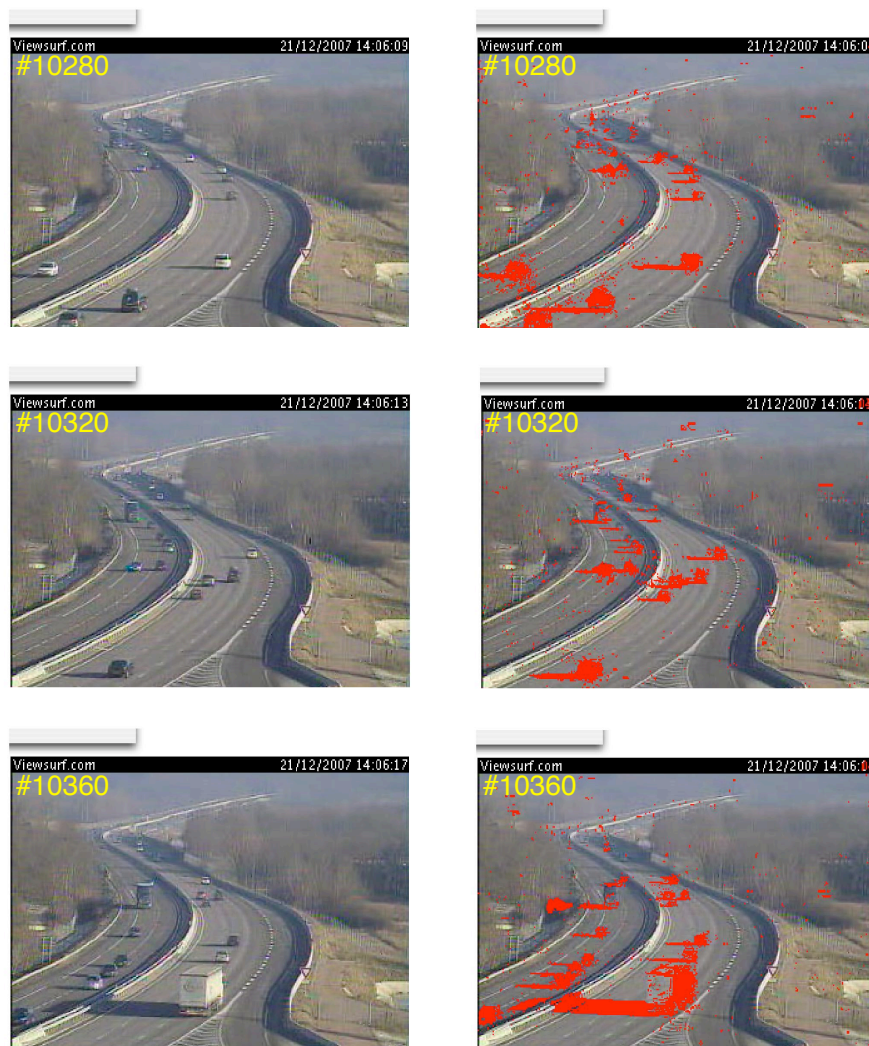


FIGURE 1.12 – Segmentation *fond / forme* dans l'espace couleurs RGB, de véhicules sur autoroute. A gauche : images brute. A droite : les pixels classifiés comme *forme* sont coloriés en rouge.

**Effet de perspective** Une autre difficulté liée au suivi visuel est due à la forte perspective provoquée par l'association d'une grande étendue de mesure (plusieurs centaines de mètres en suivi de trafic routier), et de la faible altitude des caméras. Cette perspective provoque des variations considérable de la surface apparente des objets : les objets lointains concernent beaucoup moins de pixels que les objets proches, comme illustré par les figures 1.12 et 1.14, et 1.15 : on atteint facilement un rapport de surfaces de 1 à 100. Il est alors difficile d'observer équitablement les objets lointains et les objets proches de la caméra. Nous exposons dans ce mémoire un modèle d'observation permettant de suivre les objets sur de grandes étendues de mesure, et de gérer leurs occultations, présentée dans (BC08a). Après segmentation binaire *fond / forme* de l'image, le Filtre Particulaire propose, évalue, et propage des configurations multi-objets candidates. L'évaluation est basée sur la coïncidence des projection des modèles géométriques 3D des objets, avec les pixels d'avant-plan.



FIGURE 1.13 – Segmentation *fond / forme* dans l'espace couleurs RGB, de piétons dans un centre commercial. A gauche : images brute. A droite : les pixels classifiés comme *forme* sont coloriés en rouge.



FIGURE 1.14 – Variations de la surface apparente d'un même objet au cours du temps.





FIGURE 1.15 – Des objets de surfaces apparentes très disparates.

#### 1.8.4 Modèle d'Observation Individuel, Modèle d'Observation Global

En suivi multi-objet, on trouve deux catégories de modèles d'observations : Le Modèle d'Observation Individuel délivre une réponse pour chaque objet présent dans la scène. C'est le cas des systèmes de suivi qui prennent comme entrée la sortie d'un détecteur d'objets. Khan *et al.* par exemple, disposent d'observations individuelles en entrée de leur moteur de suivi, délivrées par un détecteur d'objets (KBD04). Cette approche convient bien dans le cas d'objets à mobilité plane, tels que les fourmis qu'ils ont choisi pour illustrer leur application, mais un détecteur d'apparence d'objets est vite mis en défaut en cas d'occultations. La deuxième approche consiste à utiliser un Modèle d'Observation Global. Par définition, ce modèle ne délivre pas une information par objet. La figure 1.10 illustre un tel modèle d'observation globale, issue d'une segmentation *fond / forme*. Une telle segmentation est possible lorsque la caméra est statique, et si on peut construire un modèle d'apparence du *fond* de la scène, supposé vide de *forme* (les objets mobiles). Le modèle d'observation choisi pour cette illustration est une version simplifiée de celui exposé en section 4.4.4 : la convolution de l'image par un noyau, ici un noyau rectangulaire constant de taille arbitrairement fixée à  $20 \times 20$  pixels. Il est clair que dès qu'il y a des occultations, cette observation globale ne peut être partitionnée pour affecter un score individuel à chaque objet. D'autre part, dans cette approche, il n'y a donc pas de détecteur en amont du moteur de suivi, et on peut considérer que l'étape de détection d'objet lui est intégrée.

## 1.9 Mobilité et paramétrage des objets

**Mobilité des objets.** Il est nécessaire de distinguer les applications de suivi d'objets selon le degré de mobilité de ces derniers, *i.e.* selon le nombre de paramètres nécessaires pour caractériser la position d'un objet. Un objet rigide disposant d'une mobilité spatiale, c'est à dire libre de se déplacer dans l'espace, a un degré de mobilité de 6 (3 paramètres de translation, 3 paramètres de rotation). Un objet rigide disposant d'une mobilité plane, c'est à dire contraint de se déplacer dans un plan, a un degré de mobilité de 3 (2 paramètres de translation, 1 paramètre de rotation). Le travail présenté dans ce mémoire se limite délibérément aux objets à mobilité plane, afin que le suivi soit réalisable dans les cas où une seule caméra est disponible. Tout objet évoluant sur le sol ou à une altitude constante répond à cette contrainte. Par contre nous n'adressons pas le suivi d'objets à mobilité spatiale, tel qu'objets volants par exemple. Ce choix est dicté par l'observabilité des objets : le suivi d'objets à mobilité spatiale et de taille *a priori* inconnue impose au moins l'usage de la stéréo-vision, à cause de l'effet de projection. Sous cette condition cependant, la méthode présentée ici peut aisément être généralisée au suivi d'objets à mobilité spatiale.

**Paramétrage des objets.** Dans notre cas du suivi d'objets à mobilité plane, deux systèmes de paramétrage de leur position sont possibles : on peut paramétrer les objets soit dans le repère image, soit dans le repère réel. Le paramétrage des objets dans le repère image est le plus simple à mettre en œuvre, puisqu'il ne prend pas en compte les effets de projection et de perspective. Pour cette raison, ce paramétrage est très utilisé (DVP00; PCJM02). Il est bien adapté dans le cas d'objets se déplaçant dans un plan parallèle au plan du capteur de la caméra, ou à défaut, lorsque les effets de perspective sont négligeables (lorsque la caméra est très éloignée de la scène). Dans certaines applications, cette condition est respectée, et le suivi des objets se ramène au suivi 2D de leurs projections dans l'image. C'est le cas par exemple en surveillance aérienne du trafic routier. Dans le cas général, on ne peut pas placer la caméra en altitude, et les effets de perspective sont très marqués. Les objets sont alors obligatoirement paramétrés dans le repère réel, c'est à dire avec le nombre de paramètres correspondant au degré de mobilité des objets. par opposition au système de paramétrage précédent, nous désignerons celui ci comme *suivi 3D*. Dans le cas du suivi multi-objets, ce choix améliore considérablement la robustesse du suivi, car il permet de modéliser rigoureusement la dynamique des objets, la projection perspective, et les occultations, ce qui contraint de façon réaliste la région de l'espace d'état à explorer. Ces avantages sont notamment argumentés et exploités dans (ZN04; OWWY06; FBLF08; YO08).

**Calibration pour le suivi 3D** L'état de la configuration multi-objets est donc décrit dans le repère 3D, tandis que l'observation est décrite dans le repère 2D du plan du capteur, où se projette la scène. Le passage de l'un à l'autre de ces deux repères se fait par la matrice de projection. Celle-ci peut être calculée à partir des paramètres intrinsèques et extrinsèques de la caméra s'ils sont connus. Or ils ne sont pas toujours accessibles. C'est le cas par exemple des vidéos de surveillance de lieux publics par webcams. Il faut alors estimer directement la matrice de projection à partir de points de l'image dont les coordonnées 3D sont connues.

## 1.10 Suivi Disjoint, Suivi Joint

Le choix de l'espace d'état est un point crucial du suivi multi-objets. Deux approches sont possibles : suivi multi-objets disjoint, suivi multi-objets joint. La première approche consiste à instancier un moteur de suivi à chaque nouvel objet, qui lui sera affecté pendant tout le temps où il est perceptible. Chacun des moteurs de suivi est alors disjoint, car il suit sa propre cible sans tenir compte des autres. Dans cette approche, chaque moteur opère dans le sous-espace disjoint représentant chaque objet. La difficulté de cette approche est qu'elle demande de segmenter l'observation selon chaque objet, c'est à dire de décomposer la vraisemblance globale comme un produit de vraisemblances individuelles indépendantes :

$$p(\mathbf{Z}_t | \mathbf{X}_t) = \prod_{j=1}^J p(\mathbf{Z}_t^j | \mathbf{X}_t^j), \quad (1.9)$$

où  $p(\mathbf{Z}_t^j | \mathbf{X}_t^j)$  est la vraisemblance individuelle de l'observation  $\mathbf{Z}_t^j$  de l'objet  $j$ , étant donné son état  $\mathbf{X}_t^j$ . Cette segmentation n'est possible qu'en absence d'occultation. L'indépendance des vraisemblances individuelles n'est pas non plus valide en présence d'occultation. En vision, les occultations mutuelles entre objets rendent interdépendantes les observations des objets. Ceci génère des interactions inter-objets très difficiles à gérer. Le maintien de l'identité de chaque objet est également problématique. Dans une approche jointe, par contre, un unique moteur de suivi est en charge de la totalité des objets, il faut donc représenter tous les objets dans un espace d'état joint. Le moteur de suivi doit également gérer leurs entrées et leurs sorties. Le vecteur d'état décrit alors le nombre d'objets présents, leurs positions, vitesses, formes, catégories, identités... La nécessité de mener le suivi dans un espace joint a déjà largement été démontrée (KBD04; ZN04). Nous souhaitons néanmoins en présenter une illustration ci-après. La figure 1.16 illustre le suivi de deux cibles rectangulaires en images de synthèse binaire 100 x 100 pixels, objets en blanc, fond en noir. La cible numéro 2 est parfaitement segmentée, tandis que seuls 60% des pixels de la cible numéro 1 sont classifiés comme *forme*. Les estimations résultant du suivi sont tracées en rouge. Les deux méthodes ont les mêmes performances tant que les deux cibles n'interagissent pas, comme le montrent les images #10030. En suivi disjoint (ligne du haut), le filtre chargé du suivi de la cible #1 est attiré vers la cible #2 dès l'image #10038, car celle-ci est mieux segmentée. Le suivi de la cible #1 ne survit pas à cette occultation très partielle. Après occultation, la cible #1 n'est plus suivie (voir image #10062) ! Au contraire, le suivi joint de ces deux objets par un unique moteur de suivi, est capable de maintenir le suivi pendant et après cette occultation, comme l'illustre la ligne du bas de la figure 1.16. La fréquence de

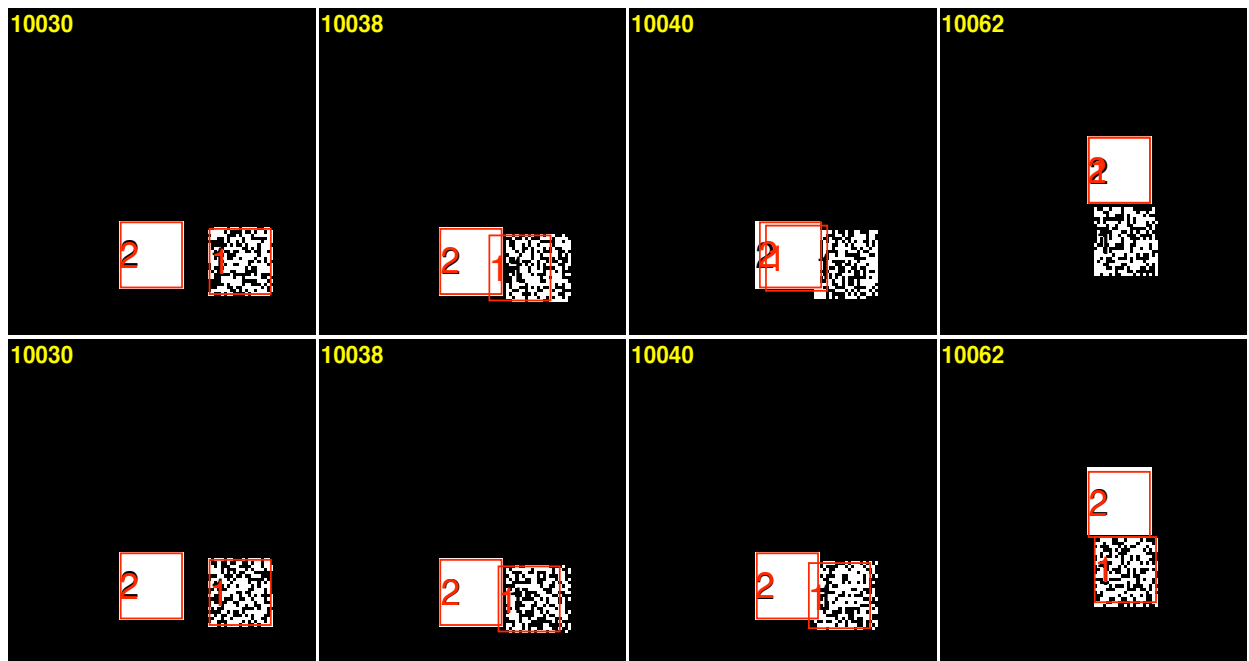


FIGURE 1.16 – Suivi de deux cibles rectangulaires de synthèse sur image binaire  $100 \times 100$  pixels, objets en blanc, fond en noir. La cible numéro 2 est parfaitement segmentée, tandis que seuls 60% des pixels de la cible numéro 1 sont classifiés comme *forme*. Estimations en rouge. En haut : suivi disjoint, un filtre instancié pour chaque cible, 20 particules par filtre. En bas : suivi joint de toutes les cibles par un seul filtre, 40 particules.

ce dysfonctionnement croît naturellement avec la densité des projections des objets dans l'image. Une expérimentation plus complète est proposée en annexe A : 10 objets sont simultanément suivis, et la précision est quantifiée, ce qui permet de comparer les approches jointe et disjointe. Le travail présenté dans les chapitres suivants repose sur une approche jointe, c'est à dire sur un unique moteur de suivi opérant dans l'espace d'état joint. Ce moteur de suivi gère les interactions entre objets, ainsi que leurs entrées et sorties. Le suivi d'un seul objet rectangulaire 2D illustré par la figure 1.16, est mené dans un espace de dimension 4 : position horizontale,

verticale, hauteur, largeur. La nécessité d'opérer dans l'espace d'état joint est lourde de conséquences, car le suivi de  $p$  objets rectangulaires  $2D$  nécessite un espace de dimension  $4p$ . Dans les applications de suivi multi-objets, un tel choix conduit à une dimension élevée de l'espace d'état. Les applications réelles telle que le suivi d'un piéton marchant sur un sol plan sont encore plus exigeantes : un vecteur d'état représentant simplement un piéton par un parallélépipède, est de dimension 8 : trois composantes pour la forme (hauteur, largeur, profondeur), trois composantes pour la position (position en  $x$ , position en  $y$ , orientation selon un axe vertical) s'il évolue sur un sol plan, deux composantes pour la vitesse. Supposons que  $N$  échantillons réalisent un bon échantillonnage de son espace d'état. Si l'on souhaite maintenant suivre  $p$  piétons, la dimension de l'espace d'état joint représentant la scène devient  $8p$ , et le nombre d'échantillons requis pour réaliser un échantillonnage de qualité identique est  $N^p$ . On voit qu'en suivi multi-objets, on va rapidement devoir travailler dans des espaces d'état de dimensions élevées, requérant un nombre d'échantillons incompatible avec les capacités des ordinateurs actuels. Les stratégies permettant de surmonter ce problème sont présentées au chapitre 2.

## 1.11 Conclusion : positionnement de la thèse

Nous avons présenté dans ce chapitre les grandes typologies des systèmes de suivi d'objet. Résultant des contraintes énoncées en section 1.1, et de nos choix, le positionnement du travail présenté dans ce mémoire est le suivant, selon ces typologies :

- ▷ Capteurs utilisés : nous avons choisi la vision pour sa grande étendue de mesure et ses faibles coûts et contraintes d'installation. Nous avons conçu un moteur de suivi prenant en compte l'observation émise par un nombre quelconque de caméras statiques, c'est à dire apte à fonctionner aussi bien en monovision, qu'en multi-vues. Nous nous limitons néanmoins au cas du suivi d'objets contenus dans une zone incluse dans l'intersection des champs de vision des caméras. Ce modèle d'observation basé sur une segmentation de l'image en *avant-plan* et *arrière-plan*, est défini au chapitre 4.
- ▷ Rapport au temps et traitement des observations : Le travail présenté dans les chapitres suivants est un système de suivi causal séquentiel, ce choix étant dicté par le besoin du temps réel.
- ▷ Suivi d'objets à mobilité plane (objets se déplaçant sur un plan), ce qui permet de réaliser le suivi même si on ne dispose que d'une caméra.
- ▷ L'hypothèse de mobilité plane permet aussi de choisir de mener le suivi en  $3D$ , c'est à dire dans l'espace matériel et non dans l'image. Ce choix permet de modéliser rigoureusement la dynamique des objets, la projection perspective, et les occultations. Ces modèles sont présentés au chapitre 4.
- ▷ Nature du modèle d'observation : Les applications visées dans ce manuscrit reposent sur des observations délivrées par des caméras statiques, ce qui permet de recourir à un modèle d'observation basé sur une segmentation de l'image courante en deux classes de pixels : pixels de *fond*, ou *arrière-plan* de l'image, et pixels de *forme*, c'est à dire tout ce qui n'a pas été reconnu comme *arrière-plan*. Les objets cibles (véhicules ou piétons) sont modélisés en  $3D$  par des parallélépipèdes. Les fortes variations de l'observabilité des cibles, dues aux occultations entre objets, aux effets de la perspective dus à la grande étendue de suivi, et à la présence simultanée de cibles de dimensions très différentes, sont modélisées telles que présentées au chapitre 4.
- ▷ La méthode retenue est probabiliste, car cette approche délivre une estimation de toute la loi de probabilité *a posteriori* de cet état. Ceci afin de prendre en compte les éventuelles multimodalités causées par les occultations et d'une façon générale par une observation bruitée.
- ▷ La représentation des lois de probabilité est non paramétrique, donc échantillonnée, car nous ne pouvons faire d'hypothèses sur les densités de probabilités. Nous sommes ainsi dans le champ des Filtrés Particulaires, dont plusieurs versions seront exposées au chapitre 2. Leurs performances seront présentées au chapitre 3.

- ▷ Afin de gérer les interactions et occultations entre objets, nous avons choisi de confier le suivi de la configuration jointe multi-objets à un unique moteur de suivi opérant dans l'espace d'état joint. Ceci permet au filtre d'estimer et gérer les interactions entre objets, ainsi que leurs entrées et sorties, comme détaillé au chapitre 4.
- ▷ Les applications de surveillance du trafic routier mettent en scène différents types de véhicules. Nous mettons en évidence dans les chapitres 4 et 5 l'intérêt de demander au filtre de mener conjointement le suivi et la classification des objets, plutôt que de mener séquentiellement ces deux opérations.
- ▷ Dans les applications de surveillance en extérieur, on est généralement confronté à des conditions d'éclairage variables au cours du temps et susceptibles de générer des ombres portées qui perturbent fortement l'observation. Nous mettons également en évidence dans les chapitres 4 et 5 l'intérêt de confier au filtre l'estimation de la source d'éclairage conjointement au suivi des objets.





---

# MÉTHODES D'ÉCHANTILLONNAGE POUR FILTRES PARTICULAIRES

---

Dans ce chapitre, après avoir rappelé le principe des Filtres Particulaires, nous nous focalisons sur la question de l'échantillonnage des particules. Il s'agit à chaque instant de les choisir judicieusement, pour représenter au mieux la loi de probabilité de l'état que l'on veut estimer, connaissant la séquence d'observation. Ce choix détermine la performance du filtre, en particulier dans les applications requérant le temps réel, et lorsque l'état à estimer se représente dans un espace de grande di-

mension. C'est le cas du suivi de multiples objets rigides. Nous examinons ensuite deux familles de filtres particuliers permettant d'opérer dans un espace de grande dimension : les Filtres Particulaires Partitionnés, et les Filtres Particulaires par Chaîne de Markov (*FP MCMC*). Pour définir ces derniers, nous examinons préalablement les échantillonneurs par Chaîne de Markov (*MCMC*). Nous proposons à la fin du chapitre un nouvel algorithme de Filtrage Particulaire *MCMC*, le *Filtre Particulaire MCMC Parallélisé*, adapté aux calculateurs parallèles.

---

## 2.1 Introduction

La fonction d'un filtre est de produire une information temporellement plus cohérente que les mesures qui l'alimentent. Cette fonctionnalité permet par exemple de maintenir une estimation vraisemblable lorsque l'observation défaille pendant quelques périodes. Bien que ce chapitre soit focalisé sur le filtrage de l'état de systèmes dynamiques, une section sera consacrée à l'échantillonnage de l'état de systèmes stationnaires. On verra en effet que cet échantillonnage est une des étapes d'un Filtrage Particulaire. Comme le Filtre de Kalman, les Filtres Particulaires appartiennent à la classe des filtres Bayésiens récurrents. Isard *et al.* ont montré qu'en suivi visuel, le Filtre de Kalman est mis en défaut notamment par un fond encombré générant des maxima locaux dans la fonction d'observation, qui « distraient » le *traqueur* de sa cible (MA98). Les Filtres Particulaires permettent de surmonter cette difficulté, et leur usage est très répandu dans le domaine du suivi d'objets, parce qu'ils gèrent les non-linéarités et multi-modalités induites par les occultations et les arrière-plans encombrés.

Les Filtres Particulaires appartiennent aussi au champ des méthodes de Monte-Carlo, car ils recourent à un tirage probabiliste d'échantillons pour approximer la loi de probabilité *a posteriori* générée par l'équation du filtre Bayésien récurrent (1.7). La clé de voûte des Filtres Particulaires est le choix judicieux de ces échantillons (les « particules ») au fil du temps. Comme l'état du système évolue au cours du temps, les échantillons doivent être déplacés vers les régions de l'espace d'état les plus porteuses d'information. Cette opération de ré-échantillonnage adaptatif doit être menée périodiquement. Rubin a proposé d'appliquer l'algorithme *SIR* (Sequential Importance Resampling Particle Filter ou Filtre Particulaire à Rééchantillonnage par Importance Séquentiel) pour approximer la loi de probabilité *a posteriori* dans un système dynamique (Rub88). Une synthèse des algorithmes de Filtrage Particulaires et de leurs fonctionnalités a été publiée par Arulampalam *et al.* dans (AMGC02). Dans le domaine du suivi visuel d'objets, Isard *et al.* (MA98) ont proposé l'algorithme *CONDENSATION*, application de l'algorithme *SIR*.

Un suivi monoculaire multi-objet basé sur le *SIR* a été proposé (IM01), et de nombreux autres ouvrages ont suivi. L'inconvénient du *SIR*, montré par de nombreux auteurs (IM01; Smi07), est qu'il est inefficace dans les espaces d'état de grande dimension, car le nombre de particules requises croît exponentiellement avec la dimension de l'espace d'état. Ainsi, un Filtre Particulaire *SIR* ne peut suivre plus de 2 ou 3 objets. Il convient donc de modifier la stratégie pour contourner cette difficulté. Parmi les stratégies proposées, nous en avons examiné et implémenté deux : les Filtres Particulaires Partitionnés proposés par Mac Cormick et Blake (MB99), et les Filtres Particulaires par Chaîne de Markov (*FP MCMC*) à propositions marginalisées au sous-espace décrivant un objet, proposés et validés sur le suivi simultané de 20 fourmis par (KBD04), puis utilisés dans le cadre du suivi de personnes par (Smi07). Une étude comparative de leurs performances est développée dans le chapitre 3. Si le Filtre Particulaire Partitionné n'échantillonne pas équitablement les objets (Smi07), il a par contre l'avantage d'être facilement parallélisable. Pour le suivi en temps réel, les *FP MCMC* ont démontré leur efficacité, permettant de suivre 2 à 4 piétons (Smi07). Plus d'objets (jusqu'à 20 fourmis) (KBD05) ont été suivis avec succès dans un cas où on évite les occultations (vue du dessus de fourmis marchant sur un sol plan). Malheureusement, la structure de son algorithme étant typiquement séquentielle, son implémentation en vue du temps réel est problématique, car il ne peut être parallélisé tel quel. Pour surmonter cette difficulté, nous proposons à la fin du chapitre une approximation du *FP MCMC* adaptée aux calculateurs parallèles.

## 2.2 Échantillonnage de Lois de Probabilité Stationnaires

### 2.2.1 Approximation d'une Loi de Probabilité par Échantillonnage

Comme illustré par la figure 2.1, une loi de probabilité de l'état  $\mathbf{X}$  d'un système quelconque, peut être approximée par une somme de  $N$  échantillons non pondérés :

$$p(\mathbf{X}) \approx \frac{1}{N} \sum_{n=1}^N \delta(\mathbf{X} - \mathbf{X}^n), \text{ que l'on note également : } p(\mathbf{X}) \approx \{\mathbf{X}^n\}_{n=1}^N \quad (2.1)$$

où  $\delta$  est l'impulsion de Dirac. Comme illustré par la figure 2.2, la même distribution peut également être approximée par une somme de  $N$  échantillons pondérés de leurs poids  $\pi^n$ ,  $n \in 1 \dots N$ , tels que  $\sum_{n=1}^N \pi^n = 1$ ,

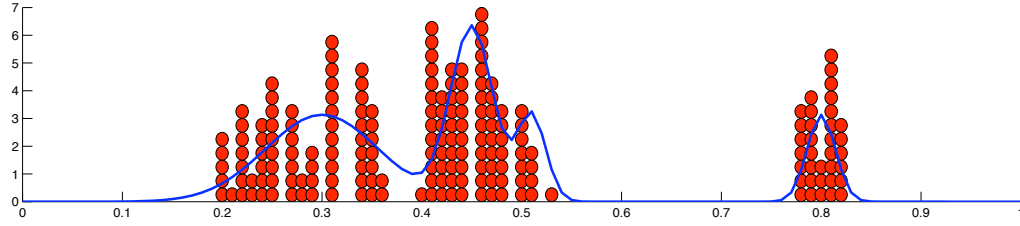


FIGURE 2.1 – Approximation d’une loi de probabilité (courbe continue bleue) par une population de  $N$  échantillons non pondérés (boules rouges).

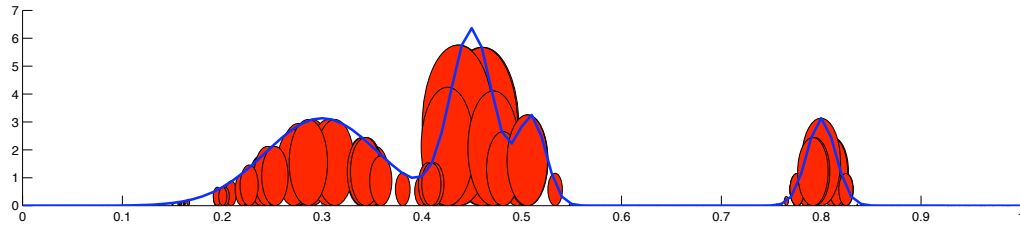


FIGURE 2.2 – Approximation d’une loi de probabilité (courbe continue bleue) par une population de  $N$  échantillons pondérés (ellipses rouges, dont la surface est proportionnelle au poids).

selon l’équation 2.2. Ces deux représentations sont deux approximations duales de la même loi de probabilité, où les nombres de copies de chaque particule dans la représentation non pondérée correspondent aux poids des particules dans la représentation pondérée.

$$p(\mathbf{X}) \approx \sum_{n=1}^N \pi^n \delta(\mathbf{X} - \mathbf{X}^n), \text{ que l'on note également : } p(\mathbf{X}) \approx \{\mathbf{X}^n, \pi^n\}_{n=1}^N \quad (2.2)$$

### 2.2.2 Choix des Échantillons

Un échantillonnage uniforme, couvrant tout l’espace d’état, est déjà peu efficace en dimension 1, comme l’illustre le graphe (a) de la figure 2.3. Un échantillonnage aléatoire est illustré graphe (b). Il ne fait pas mieux, et les deux graphes ont en commun un important gaspillage d’échantillons qui tombent dans des régions de l’espace où il ne se passe rien. L’échantillonnage adaptatif réalisé sur le graphe (c) est beaucoup plus intéressant : il ajuste la densité des échantillons en fonction de l’intérêt de la région. Dans les applications de suivi multi-objets, la dimension de l’espace d’état est élevée : une application typique est le suivi d’un piéton marchant sur un sol plan. Son descripteur le plus simple (un cylindre) nécessite un vecteur de dimension 6 : deux composantes pour la forme (rayon et hauteur), deux composantes pour la position, deux composantes pour la vitesse. Supposons que  $N$  échantillons réalisent un bon échantillonnage de son espace d’état. Si l’on souhaite maintenant suivre  $p$  piétons, la dimension de l’espace d’état joint représentant la scène devient  $6p$ . Le nombre d’échantillons nécessaires devient  $N^{6p}$ . Cette croissance exponentielle selon la dimension de l’espace rend la méthode inopérante dès qu’il y a plus de deux objets à suivre (IM01). L’échantillonnage adaptatif réalisé sur le graphe (c) devient alors indispensable.

## 2.3 Échantillonnage de Lois Dynamiques : Filtres Particulaires

L’équation du filtre Bayésien récursif (1.7) définit la loi de probabilité de l’état d’un système dynamique à l’instant courant, à partir de la loi de probabilité de son état à l’instant précédent, et de l’observation courante.

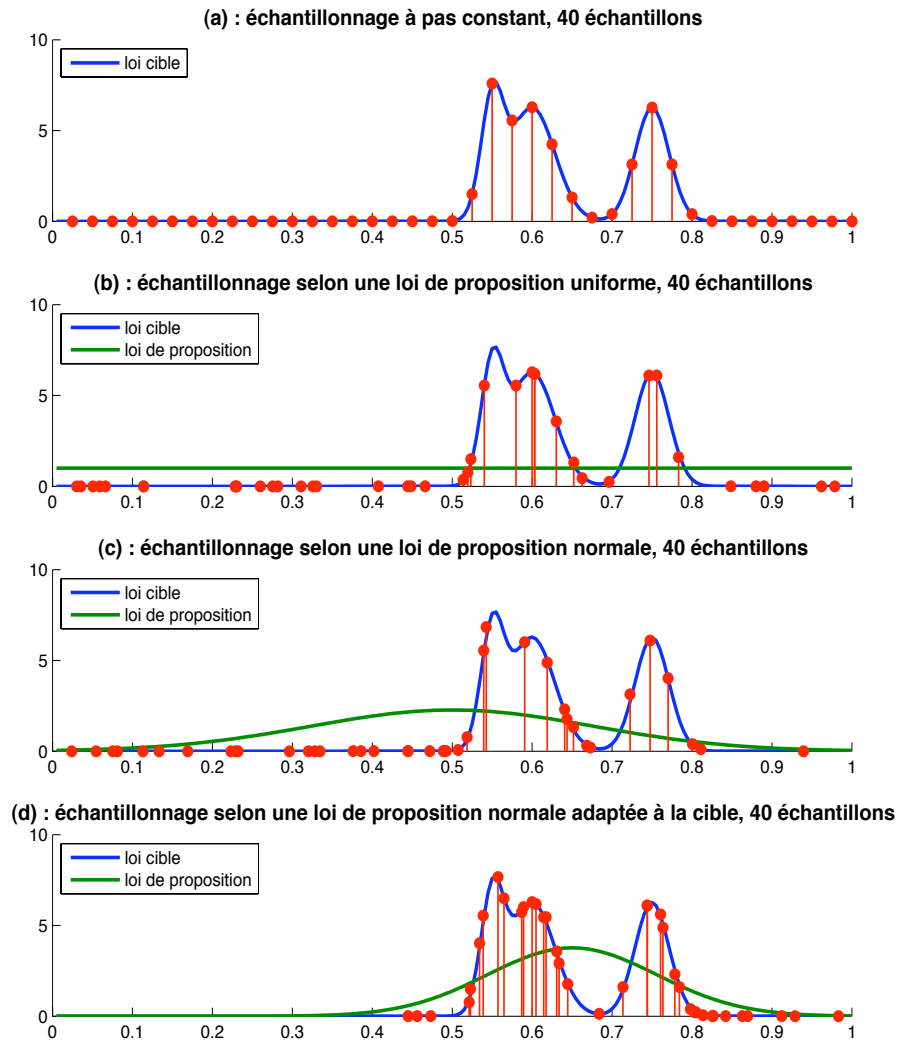


FIGURE 2.3 – Échantillonnage d’une loi stationnaire (courbe continue bleue) par une population de  $N = 40$  échantillons (points rouges), tirés d’une loi de proposition inadaptée à la cible ((a), (b), et (c)). On obtient évidemment un échantillonnage plus efficace si la loi de proposition est adaptée à la cible (c’est à dire qu’on a une petite idée de celle-ci), tout en permettant des tirages faciles (ici  $\mathcal{N} = (0.65, \sigma = 0.15)$ ).

Malheureusement, l'intégrale qu'elle contient ne peut en général être calculée analytiquement. On est alors obligé de recourir à l'échantillonnage pour approximer ces lois de probabilités. Les Filtres Particulaires produisent une approximation échantillonnée de cette intégrale. L'efficacité d'un Filtre Particulaire dépend de sa capacité à exploiter les données antérieures, pour choisir au mieux les échantillons.

### 2.3.1 Filtres Particulaires : typologies

Une première famille de stratégies, les Filtres Particulaires Parallèles, est utilisée avec succès en vision depuis plusieurs années. La méthode est basée sur l'algorithme *SIR* (Sequential Importance Resampling) ou *CONDENSATION* proposé par Isard et Blake (MA98) : on lance en parallèle des hypothèses indépendantes, on les évalue en parallèle, et on propage les meilleures à l'observation suivante. Cette stratégie est intéressante car elle permet la parallélisation des calculs les plus lourds en suivi visuel : l'évaluation de la vraisemblance des hypothèses connaissant une observation lourde à traiter (une image). Les stratégies de la deuxième famille, les Filtres Particulaires *MCMC* (ou *MCMC PF* : Markov Chain Monte-Carlo Particle Filters) (Mac03), génèrent une chaîne de Markov d'hypothèses, la transition d'une hypothèse à la suivante étant conditionnée par l'observation. Le processus d'exploration de l'espace à un instant  $t$  est donc chaîné, ou itératif : il est plus « informé » que dans les Filtres Particulaires Parallèles, puisque l'observation intervient dans la transition entre deux hypothèses. C'est son point fort, et l'utilisation des Filtres Particulaires *MCMC* en suivi visuel, bien que plus récente, a montré un potentiel certain (KBD04).

### 2.3.2 Filtres Particulaires *SIR*

On a vu l'approximation échantillonnée d'une loi stationnaire en section 2.2. Dans ce cas, on n'a en général pas d'*a priori* sur cette loi. Dans le cas d'une loi dynamique, on peut s'appuyer sur une prédiction générée à partir du passé et d'un modèle dynamique du système. C'est ce que proposent les Filtres Particulaires : ils propagent temporellement une approximation échantillonnée de la loi dynamique. Les Filtres Particulaires reposent sur l'équation du filtre Bayésien récursif (1.7), qui définit la loi de probabilité de l'état d'un système dynamique à l'instant courant, à partir de la loi de probabilité de son état à l'instant précédent, et de l'observation courante. Comme l'intégrale qu'elle contient ne peut en général pas être calculée analytiquement, on recourt à l'échantillonnage pour approximer ces lois de probabilités. Les Filtres Particulaires produisent donc une approximation échantillonnée de cette intégrale. L'efficacité d'un Filtre Particulaire dépend de sa capacité à exploiter les données antérieures, pour choisir au mieux les échantillons. Les Filtres Particulaires *SIR* (Sequential Importance Resampling) ainsi que les Filtres Particulaires *SIS* (Sequential Importance Sampling) sont décrits en annexe B. Ils s'appuient sur une étape d'*Échantillonnage préférentiel* ou *IS* (Importance Sampling), également décrit en annexe B.

Supposons que l'on dispose d'une approximation de  $p(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1})$ , loi de probabilité *a posteriori* de l'état à l'instant  $t - 1$  par  $N$  échantillons discrets pondérés  $\{\mathbf{X}_{t-1}^n, \pi_{t-1}^n\}_{n=1}^N$  :

$$p(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1}) \approx \sum_{n=1}^N \pi_{t-1}^n \delta(\mathbf{X}_{t-1} - \mathbf{X}_{t-1}^n), \quad (2.3)$$

où  $\delta$  est l'impulsion de Dirac, et  $\pi_{t-1}^n$  est le poids du  $n^{\text{ième}}$  échantillon,  $n \in 1 \dots N$ , tel que  $\sum_{n=1}^N \pi_{t-1}^n = 1$ . L'approximation échantillonnée de l'équation (1.6) s'écrit alors :

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t-1}) \approx \sum_{n=1}^N \pi_{t-1}^n p(\mathbf{X}_t|\mathbf{X}_{t-1}^n), \quad (2.4)$$

où  $p(\mathbf{X}_t|\mathbf{X}_{t-1})$  est la loi de probabilité d'évolution dynamique du système. La loi définie par l'équation (2.4) est une mixture des  $N$  composantes  $p(\mathbf{X}_t|\mathbf{X}_{t-1}^n)$ , pondérées des poids  $\pi_{t-1}^n$ . L'équation du filtre Bayésien récursif (1.7) peut alors être approximée par :

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t}) \approx C^{-1} p(\mathbf{Z}_t|\mathbf{X}_t) \sum_{n=1}^N \pi_{t-1}^n p(\mathbf{X}_t|\mathbf{X}_{t-1}^n). \quad (2.5)$$

Comme on ne dispose généralement pas non plus d'expression analytique de  $p(\mathbf{Z}_t|\mathbf{X}_t)$ , il faut à nouveau échantillonner. On pourrait bien sûr reprendre les anciens échantillons  $\mathbf{X}_{t-1}^n$ , mais ils ne sont pas forcément les plus pertinents pour représenter efficacement  $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$ , car la région intéressante de l'espace a pu changer entre  $t-1$  et  $t$ . Il faut donc rééchantillonner, c'est à dire choisir de nouveaux échantillons  $\mathbf{X}_t^n$ . Le *SIR* est décrit par l'algorithme 2.1. Les étapes (a) et (b) réalisent respectivement un échantillonnage par importance dans le nuage de particules pondérées  $\{\mathbf{X}_{t-1}^n, \pi_{t-1}^n\}_{n=1}^N$ , et un tirage dans la loi dynamique appliquée à chacun de ces échantillons. Ces deux opérations réalisées séquentiellement, équivalent à tirer  $N$  échantillons d'état  $\mathbf{X}_t^n$  selon (2.6) :

$$\mathbf{X}_t^n \sim q(\mathbf{X}_t) = \sum_{n=1}^N \pi_{t-1}^n p(\mathbf{X}_t|\mathbf{X}_{t-1}^n) \quad (2.6)$$

Autrement dit, un filtre particulaire produit à chaque pas temporel  $t$  un échantillonnage par importance de la loi de probabilité *a posteriori* à l'instant  $t$ . Pour chacun des échantillons  $\mathbf{X}_t^n$  on calcule enfin la vraisemblance de l'observation :  $\pi_t^n = P(\mathbf{Z}_t|\mathbf{X}_t^n)$ . Le filtre délivre alors le lot de  $N$  échantillons discrets pondérés  $\{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N$ , approximant  $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$ , loi de probabilité *a posteriori* de l'état à l'instant  $t$ .

L'algorithme *SIR* est détaillé algorithme 2.1, et est illustré en dimension 1 par la figure 2.4. L'algorithme 2.1 montre que le nuage des particules est rééchantillonné en parallèle. L'opération de rééchantillonnage qui fait l'interêt du L'algorithme *SIR* exécute trois étapes à chaque pas temporel, à partir du nuage de particules approximant la loi de probabilité à  $t-1$  :

- ▷ (a) Tirage par Importance : tirage avec remise des particules selon leur « importance » (leur poids) selon l'observation à  $t-1$ . Cette opération remplace les particules de poids forts par de nombreuses particules, et les particules de poids faibles par peu de particules. A ce stade elles sont toutes affectées de poids identiques.
- ▷ (b) Évolution temporelle : remplacer chaque particule selon la fonction de proposition  $p(\mathbf{X}^*|\mathbf{X})$ . Si on n'a aucune information sur la loi d'évolution du processus à estimer, on peut choisir une loi normale  $p(\mathbf{X}^*|\mathbf{X}) = \mathcal{N}(0, \sigma)$ , la valeur de  $\sigma$  étant choisie avec soin car c'est elle qui détermine l'échantillonnage de l'espace d'état par les particules. Mais le filtre sera beaucoup plus performant si on inclut à ce stade un modèle d'évolution, prenant en compte la dynamique des objets suivis.
- ▷ (c) Ponderation des nouvelles particules par la vraisemblance de l'observation  $\mathbf{Z}_t$  à l'instant  $t$ , sachant l'échantillon d'état  $\mathbf{X}_t^n$  :  $\pi_t^n = P(\mathbf{Z}_t|\mathbf{X}_t^n)$

La propagation des particules de  $t-1$  à  $t$  fait la force de l'algorithme *SIR*, et lui donne naturellement les propriétés d'un filtre. A chaque itération, on attire des échantillons vers les régions de forte loi de probabilité, ce qui rend l'échantillonnage adaptatif. Le choix du modèle d'évolution temporelle des particules détermine l'efficacité du rééchantillonnage. La figure 2.5 montre qu'une trop faible amplitude dans les propositions d'évolution ( $p(\mathbf{X}^*|\mathbf{X}) = \mathcal{N}(0, \sigma = 0.02)$ ) conduit à totalement rater le pic centré sur  $X = 0.62$  à l'instant  $t$  (graphe (c)), car aucune particule n'a été envoyée dans cette région par l'étape de prédiction (b). Seules les régions de l'espace d'état où des particules ont été envoyées sont correctement estimées. A l'opposé, la figure 2.6 montre qu'une amplitude excessive des propositions d'évolution ( $p(\mathbf{X}^*|\mathbf{X}) = \mathcal{N}(0, \sigma = 0.50)$ ) conduit à disséminer les particules dans tout l'espace d'état, ce qui conduit à un sous échantillonnage des régions importantes. Ces problèmes sont amplifiés lorsqu'on échantillonne dans un espace de grande dimension. Le point faible de cet algorithme est qu'il requiert un nombre de particules, lié exponentiellement à la dimension de l'espace, défaut pointé dans de nombreuses études, parmi lesquelles (IM01; SGP04).

### 2.3.3 Rééchantillonnage par Importance Séquentiel Partitionné

L'algorithme 2.1 est mis en défaut lorsque la dimension de l'espace augmente, car on sous-échantillonne l'espace, et on risque de complètement manquer des régions de forte probabilité. L'algorithme 2.2, appelé *SIR Partitionné* (MB99), a été conçu pour surmonter cette difficulté. Comme le *SIR*, le *SIR* partitionné appartient au champ des filtres particulaires, et est également une méthode parallèle. Le Filtrage Particulaire Partitionné est

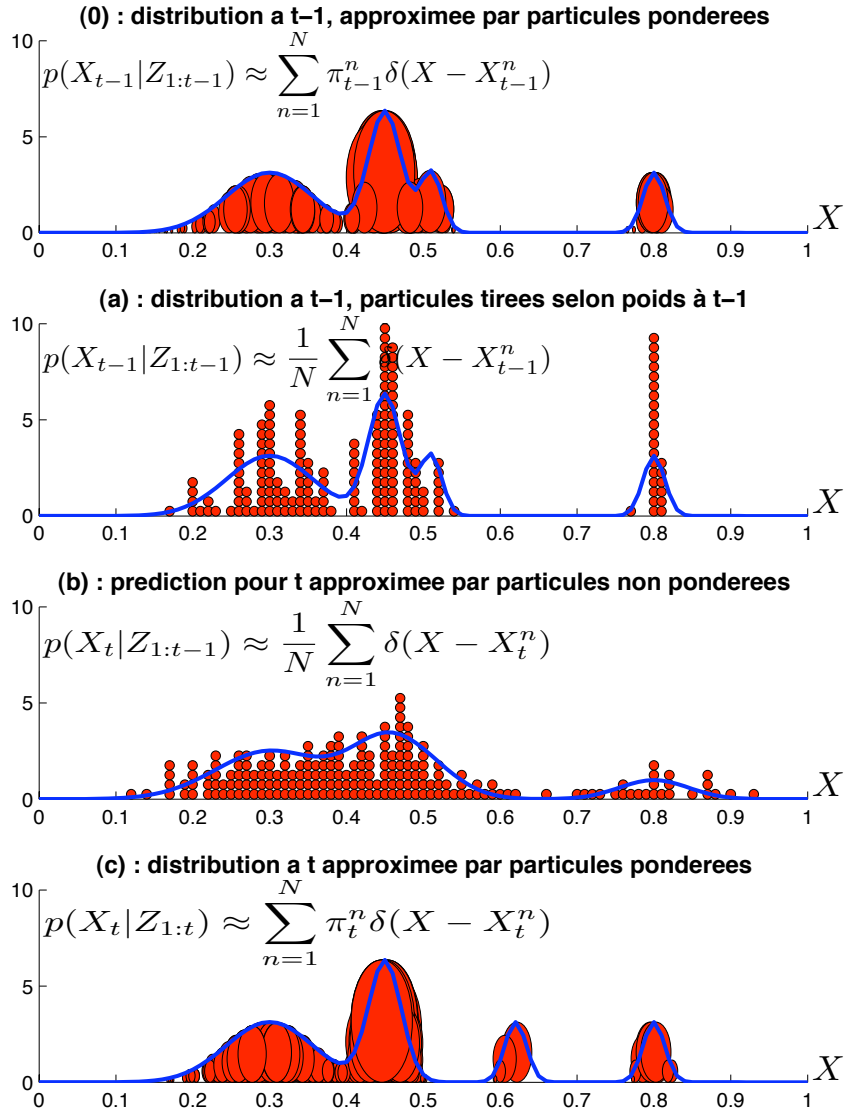


FIGURE 2.4 – Propagation des particules par l’algorithme *SIR* avec évolution dynamique des particules  $\mathcal{N}(0, \sigma = 0.06)$ . (0) : Loi de probabilité *a posteriori* à  $t - 1$  (courbe bleue) et son approximation par des particules pondérées (ellipses rouge de surface proportionnelle à leur poids). (a) : La même loi de probabilité *a posteriori* à  $t - 1$  (courbe bleue) et son approximation par des particules non pondérées, après rééchantillonnage par Importance selon le poids des particules à l’instant  $t - 1$ . (b) : Loi de probabilité *a priori* à  $t$  (courbe bleue) et son approximation par des particules non pondérées, générées selon le modèle dynamique  $\mathcal{N}(0, \sigma = 0.06)$ . (c) : Loi de probabilité *a posteriori* à  $t$  (courbe bleue) et son approximation par des particules pondérées selon l’observation à l’instant  $t$ .

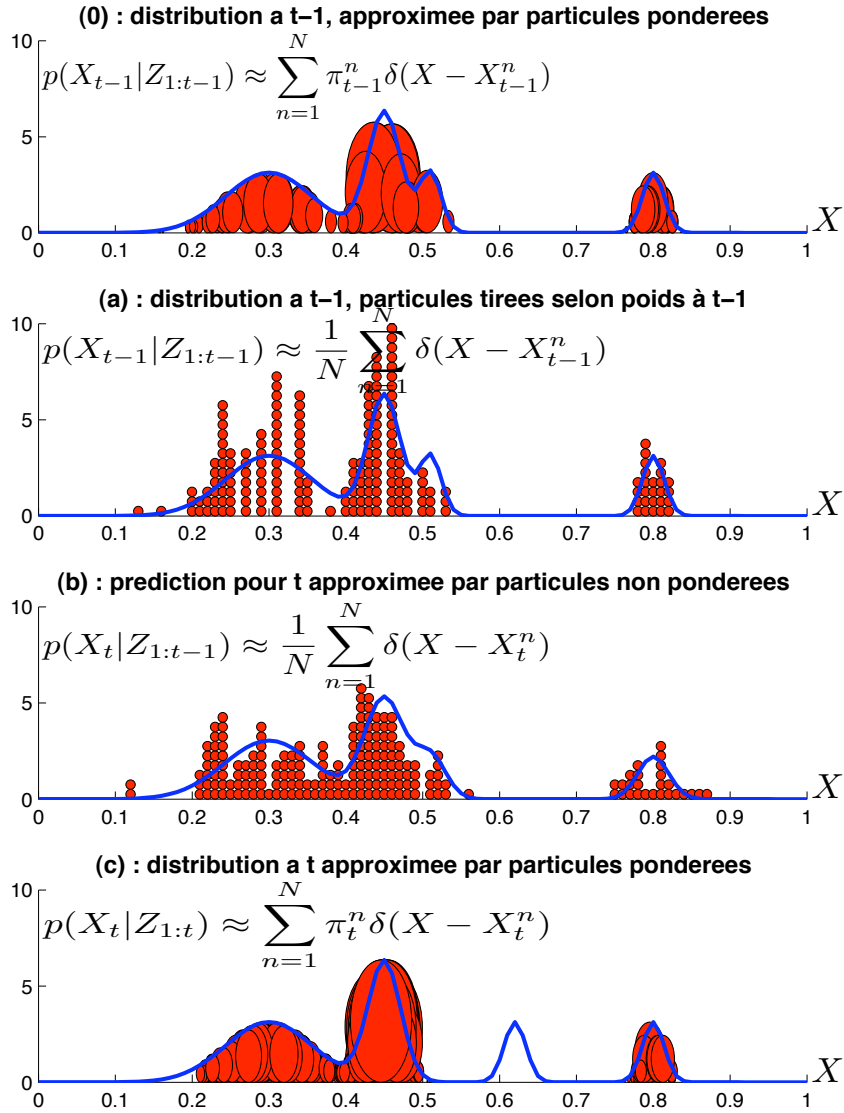


FIGURE 2.5 – Propagation des particules par l’algorithme *SIR* avec évolution dynamique des particules  $\mathcal{N}(0, \sigma = 0.02)$ . (0) : Loi de probabilité *a posteriori* à  $t - 1$  (courbe bleue) et son approximation par des particules pondérées (ellipses rouge de surface proportionnelle à leur poids). (a) : La même loi de probabilité *a posteriori* à  $t - 1$  (courbe bleue) et son approximation par des particules non pondérées, après rééchantillonnage par Importance selon le poids des particules à l’instant  $t - 1$ . (b) : Loi de probabilité *a priori* à  $t$  (courbe bleue) et son approximation par des particules non pondérées, générées selon le modèle dynamique  $\mathcal{N}(0, \sigma = 0.02)$ . (c) : Loi de probabilité *a posteriori* à  $t$  (courbe bleue) et son approximation par des particules pondérées selon l’observation à l’instant  $t$ .



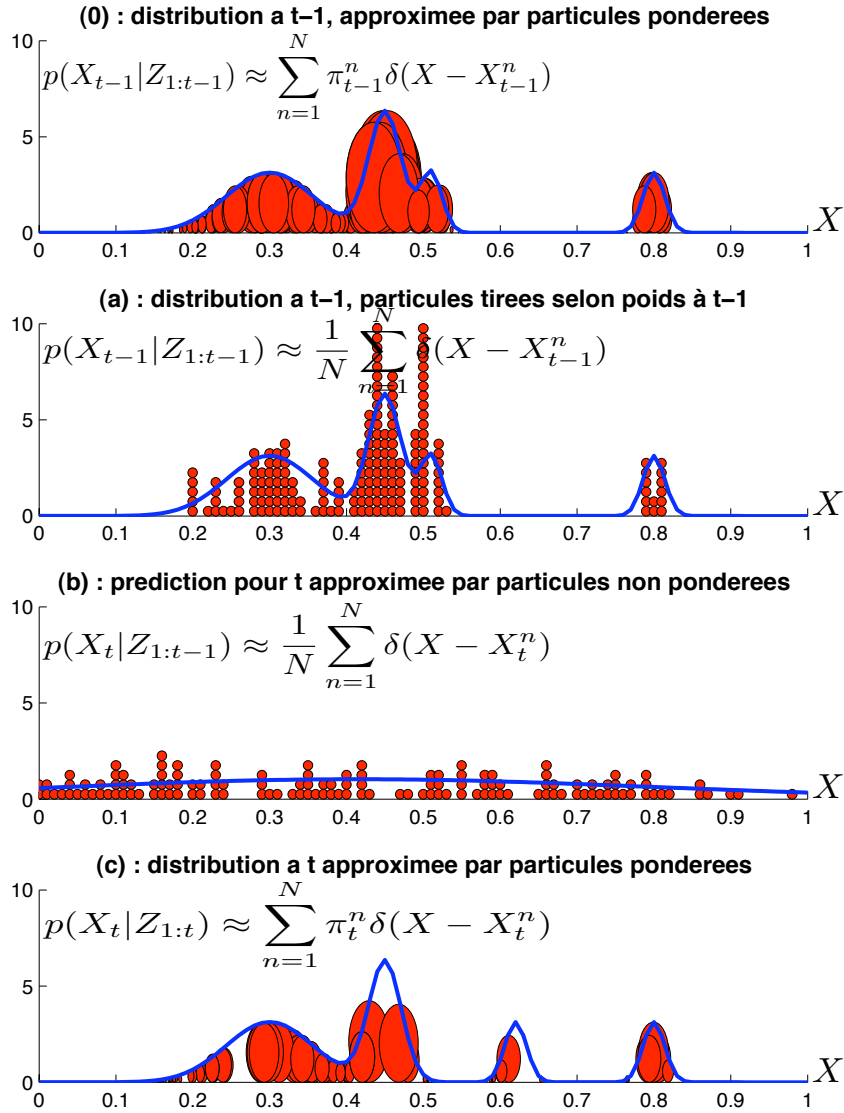


FIGURE 2.6 – Propagation des particules par l’algorithme *SIR* avec évolution dynamique des particules  $\mathcal{N}(0, \sigma = 0.5)$ . (0) : Loi de probabilité *a posteriori* à  $t - 1$  (courbe bleue) et son approximation par des particules pondérées (ellipses rouge de surface proportionnelle à leur poids). (a) : La même loi de probabilité *a posteriori* à  $t - 1$  (courbe bleue) et son approximation par des particules non pondérées, après rééchantillonnage par Importance selon le poids des particules à l’instant  $t - 1$ . (b) : Loi de probabilité *a priori* à  $t$  (courbe bleue) et son approximation par des particules non pondérées, générées selon le modèle dynamique  $\mathcal{N}(0, \sigma = 0.5)$ . (c) : Loi de probabilité *a posteriori* à  $t$  (courbe bleue) et son approximation par des particules pondérées selon l’observation à l’instant  $t$ .

**Algorithme 2.1** Rééchantillonnage Séquentiel par Importance des Échantillons (*SIR*)**Entrées :**

- nuage de particules et poids associés approximant la loi de probabilité *a posteriori* de l'état  $\mathbf{X}$  à l'instant  $t - 1$  :  $\{\mathbf{X}_{t-1}^n, \pi_{t-1}^n\}_{n=1}^N$

- loi de probabilité de prédiction  $p(\mathbf{X}_t | \mathbf{X}_{t-1})$  de l'état.

- observation  $\mathbf{Z}_t$  à l'instant  $t$ .

1. Échantillonnage par importance :

**pour**  $n = 1$  à  $N$  **faire**(a) Rééchantillonnage : tirer une particule  $\mathbf{X}_{t-1}^i$ ,  $i \in \{1, \dots, N\}$ , selon la probabilité  $\pi_{t-1}^i$ (b) Prédiction : tirer une proposition  $\mathbf{X}_t^n$  selon la loi de probabilité d'évolution  $p(\mathbf{X}_t | \mathbf{X}_{t-1})$ (c) affecter à  $\mathbf{X}_t^n$  le poids  $\pi_t^n = P(\mathbf{Z}_t | \mathbf{X}_t^n)$ , vraisemblance de l'observation  $\mathbf{Z}_t$  sachant l'état  $\mathbf{X}_t^n$ .**fin pour**2. normaliser les poids :  $\pi_t^n = \frac{\pi_t^n}{\sum \pi_t^n}$ **Sortie :** nuage de particules approximant la loi de probabilité à l'instant  $t$  :  $\{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N$ 

aussi basé sur l'estimation de la loi de probabilité *a posteriori*  $p(\mathbf{X}_t | \mathbf{Z}_{1:t})$  de l'état  $\mathbf{X}_t$  à l'instant  $t$ , conditionné à l'historique des mesures  $\mathbf{Z}_{1:t}$ , par un ensemble de  $N$  particules pondérées  $\{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N$  avec les poids associés  $\pi_t^n \propto P(\mathbf{Z}_t | \mathbf{X}_t^n)$ . Cette méthode consiste à appliquer le *SIR* au nuage de particules, préalablement attirées vers les régions intéressantes de l'espace d'état. La plus value de cette méthode réside dans cette attraction, que l'on réalise par une succession de rééchantillonnages du nuage selon des sous espaces marginalisant l'espace d'état, afin de favoriser les régions intéressantes. Cette attraction n'est pas toujours possible : elle nécessite que l'on dispose d'une loi de vraisemblance marginale de l'observation  $\mathbf{Z}_t$  sachant chacune des  $d^{\text{ièmes}}$  composantes  $x_t^d$ , notée  $p_d(\mathbf{Z}_t | x_t^d)$  dans l'algorithme 2.2. Cette condition n'est pas toujours réalisée, notamment dans les applications de suivi visuel multi-objets, notamment à cause des occultations mutuelles entre objets. L'algorithme 2.2 expose le processus dans le cas extrême, où on marginalise selon chacune des  $D$  dimensions de l'espace d'état. Cependant, la marginalisation de l'espace peut être choisie en fonction du besoin : dans le cas du suivi multi-objet, l'approche la plus fréquente consiste à associer un sous-espace à chaque objet. On évite ainsi la croissance exponentielle du nombre de particules lorsqu'on veut utiliser le *SIR* pour suivre un grand nombre d'objets. L'algorithme requiert cependant un volume de calculs proportionnel au nombre de rééchantillonnages successifs à mener, donc au nombre d'objets présent. Cette méthode a été utilisée dans le suivi de plusieurs objets opaques par (MB99). Dans cette application, les objets sont ordonnés du plus proche au plus lointain afin de modéliser leurs occultations, et chaque sous-espace décrit un objet. Le *SIR Partitionné* présente malheureusement un défaut, identifié dans (SGP04). Il réalise en effet une succession de rééchantillonnages du nuage de particules selon chaque objet, qui produit un nuage de particules contenant de nombreuses instances identiques du premier objet selon lequel on a rééchantillonné. Ceci mène à la situation paradoxale suivante : plus l'objet est proche de la caméra (donc meilleure est son observabilité), plus sa loi de probabilité est représentée par un petit nombre de clones ! Une version améliorée de la méthode (non testée ici) a été proposée pour atténuer ce défaut, le Distributed Partitionned Particle Sampling (SGP04). Elle consiste à scinder le nuage en plusieurs sous-nuages auxquels on fait subir les rééchantillonnages marginalisés selon des ordres différents, afin de ne favoriser aucun objet.

## 2.4 Échantillonneurs *MCMC*

Un échantillonneur permet d'approximer une loi de probabilité stationnaire inconnue notée  $\pi$  définie sur l'espace de la variable aléatoire  $\mathbf{X}$ , par un nuage d'échantillons non pondérés, tel que défini dans l'équation (2.1) (section 2.2.1). Lorsqu'on n'a pas de modèle simple de  $p(\mathbf{X})$ , le tirage des échantillons est difficile. Une solution très employée est le recours à l'*Échantillonnage Préférentiel* (ou *Importance Sampling*, *IS*), méthode d'échantillonnage parallèle décrit dans l'annexe B. Les échantillonneurs *MCMC* (Markov Chain Monte-Carlo) forment au contraire une famille de méthodes d'échantillonnage non parallèles mais chaîné, c'est à dire que chaque échantillon est généré par une fonction dite fonction d'importance ou de proposition, à partir de l'échantillon

**Algorithme 2.2** Rééchantillonnage par Importance Séquentiel Partitionné**Entrées :**

- nuage de particules et poids associés approximant la loi de probabilité *a posteriori* de l'état  $\mathbf{X}$  à l'instant  $t - 1$  :  $\{\mathbf{X}_{t-1}^n, \pi_{t-1}^n\}_{n=1}^N$  avec  $\mathbf{X}_{t-1}^n = [x_{t-1}^{d,n}]_{d=1}^D$ , où  $D$  est la dimension de l'espace d'état.
- lois de probabilité de dynamique  $p(x_t^d | x_{t-1}^d)$  de la  $d^{\text{ième}}$  composante.
- observation  $\mathbf{Z}_t$  à l'instant  $t$ .
- lois de vraisemblance marginales  $p_d(\mathbf{Z}_t | x_t^d)$  de l'observation  $\mathbf{Z}_t$  sachant la  $d^{\text{ième}}$  composante  $x_t^d$ .
- loi de vraisemblance jointe  $p(\mathbf{Z}_t | \mathbf{X}_t)$  de l'observation  $\mathbf{Z}_t$  sachant l'état  $\mathbf{X}_t$ .

1. Rééchantillonnage par importance selon les poids  $\pi_{t-1}^n$  :**pour**  $n = 1$  à  $N$  **faire**

- tirer une particule  $\mathbf{X}_{t-1}^i$ ,  $i \in \{1, \dots, N\}$ , selon la probabilité  $\pi_{t-1}^i$
- définir  $\{\mathbf{X}_{t-1}^{*n}, \pi_{t-1}^{*n}\} = \{\mathbf{X}_{t-1}^i, 1/N\}$

**fin pour**recopier  $\{\mathbf{X}_{t-1}^n, \pi_{t-1}^n\}_{n=1}^N = \{\mathbf{X}_{t-1}^{*n}, \pi_{t-1}^{*n}\}_{n=1}^N$ 

2. Rééchantillonnages successifs :

**pour**  $d = 1$  à  $D - 1$  (pour chaque dimension sauf la dernière) **faire**(a) prédiction selon  $d$  :**pour**  $n = 1$  à  $N$  **faire**

- tirer  $x_t^{d,*}$  de la loi de probabilité de prédiction  $p(x_t^d | x_{t-1}^{d,n})$
- remplacer  $x_t^{d,n}$  par  $x_t^{d,*}$  dans  $\mathbf{X}_t^n$

**fin pour**(b) Rééchantillonnage selon la dimension  $d$  :**pour**  $n = 1$  à  $N$  **faire**

- tirer une particule  $\{\mathbf{X}_t^i, \pi_t^i\}$ ,  $i \in \{1, \dots, N\}$ , selon la probabilité  $\pi_t^{d,i} = P_d(\mathbf{Z}_t | x_t^{d,i})$
- définir  $\{\mathbf{X}_t^{*n}, \pi_t^{*n}\} = \{\mathbf{X}_t^i, \pi_t^i / \pi_t^{d,i}\}$

**fin pour**(c) recopier  $\{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N = \{\mathbf{X}_t^{*n}, \pi_t^{*n}\}_{n=1}^N$ (d) normaliser les poids :  $\pi_t^n = \frac{\pi_t^{*n}}{\sum \pi_t^{*n}}$ **fin pour**3. prédiction selon la dernière dimension  $D$  :**pour**  $n = 1$  à  $N$  **faire**

- tirer  $x_t^{D,*}$  de la loi de probabilité de dynamique  $p(x_t^D | x_{t-1}^{D,n})$
- remplacer  $x_t^{D,n}$  par  $x_t^{D,*}$  dans  $\mathbf{X}_t^n$

**fin pour**

4. Calcul des vraisemblances sur l'espace joint :

**pour**  $n = 1$  à  $N$  **faire**affecter à  $\mathbf{X}_t^n$  le poids  $\pi_t^n \propto \pi_t^{*n} \cdot P(\mathbf{Z}_t | \mathbf{X}_t^n)$ , vraisemblance de l'observation  $\mathbf{Z}_t$  sachant l'état  $\mathbf{X}_t^n$ .**fin pour**- normaliser les poids :  $\pi_t^n = \frac{\pi_t^n}{\sum \pi_t^n}$ **Sortie :** nuage de particules approximant la loi de probabilité à l'instant  $t$  :  $\{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N$

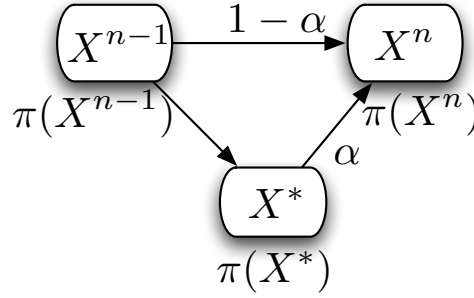


FIGURE 2.7 – Une itération de l'échantillonneur de Metropolis : une proposition  $\mathbf{X}^*$  est générée à partir de la loi  $q(\mathbf{X}^*|\mathbf{X}^{n-1})$ . Elle est acceptée avec la probabilité  $\alpha$  calculée à partir des évaluations de la loi de probabilité  $\pi$  pour les deux états :  $\pi(\mathbf{X}^{n-1})$  et  $\pi(\mathbf{X}^*)$ , sinon l'état précédent  $\mathbf{X}^{n-1}$  est dupliqué.

précédent. Nous consacrons cette section aux échantillonneurs *MCMC* pour deux raisons :

- ▷ Les algorithmes de Filtrage Particulaire *MCMC* présentés en section 2.5, mettent en oeuvre à chaque pas temporel un échantillonnage *MCMC* de la loi de probabilité de l'état *a posteriori* du système.
- ▷ Une méthode d'estimation de trajectoire de multiples objets non séquentielle fait appel à une chaîne de Markov pour explorer l'espace des trajectoires d'objets sur toute une séquence (YMC07).

### 2.4.1 Échantillonneur de Metropolis

L'échantillonneur de Metropolis est une méthode itérative ou chaînée : l'exploration de l'espace est menée par une chaîne de Markov d'ordre 1, selon l'algorithme 2.3, où la transition d'un échantillon  $\mathbf{X}^{n-1}$  au suivant  $\mathbf{X}^n$ , est assurée par l'intermédiaire d'une proposition  $\mathbf{X}^*$  tirée de la loi  $q(\mathbf{X}^*|\mathbf{X}^{n-1})$ . Cette loi n'a rien de dynamique, puisque le temps n'intervient pas ici. C'est une loi d'exploration de l'espace d'état. Le rapport des vraisemblances respectives de l'observation  $\mathbf{Z}$  sachant  $\mathbf{X}^*$  et  $\mathbf{X}^{n-1}$  détermine la probabilité d'acceptation de  $\mathbf{X}^*$ . En cas de refus  $\mathbf{X}^{n-1}$  est dupliqué. Une exploration avec l'échantillonneur de Metropolis est une marche aléatoire conduite par sa loi de proposition  $q(\mathbf{X}^*|\mathbf{X})$ . Les hypothèses ne sont donc pas indépendantes. La figure 2.7 illustre une itération de l'échantillonneur de Metropolis : la proposition  $\mathbf{X}^*$  est acceptée avec la probabilité  $\alpha$  calculée selon l'algorithme 2.3, sinon on recopie l'état précédent  $\mathbf{X}^n = \mathbf{X}^{n-1}$ . Metropolis a montré que son échantillonneur converge vers la distribution cible, à condition que la chaîne soit « suffisamment longue », pour rendre les hypothèses pseudo-indépendantes, et que la loi de proposition  $q(\mathbf{X}^*|\mathbf{X})$  soit symétrique par rapport à  $\mathbf{X}$ . En pratique, il n'est pas facile de déterminer la « longueur suffisante » de la chaîne, l'expérimentation est indispensable. Les conditions ci-dessous permettent d'accélérer la convergence :

- ▷ Un échantillon d'initialisation  $\mathbf{X}^{ini}$  pas trop décentré par rapport à la cible.
- ▷ La suppression des premiers échantillons de la chaîne, biaisés par leur dépendance à l'échantillon de départ  $\mathbf{X}^0$ . Cette opération est dénommée *Burn-in*. L'expérience montre qu'un *Burn-in* de 20% de la chaîne, c'est à dire l'élimination des premiers 20% de la chaîne rend le biais d'initialisation négligeable.
- ▷ L'adaptation de la loi de proposition  $q(\mathbf{X}^*|\mathbf{X})$  à la fonction cible (a priori inconnue). Si on n'en a aucune idée, on pourra par exemple choisir comme loi de probabilité de proposition une loi normale  $\mathcal{N}(0, \sigma)$ , mais l'exploration convergera beaucoup plus rapidement si on a une connaissance a priori de la cible. On rend alors la loi de proposition  $q(\mathbf{X}^*|\mathbf{X})$  adaptative aux observations (ce sont les DDMCMC, ou Data-Driven *MCMC*).

**Algorithme 2.3** Échantillonneur de Metropolis**Entrées :**

- particule d'initialisation  $\mathbf{X}^{ini}$
- loi de probabilité de proposition  $q(\mathbf{X}^*|\mathbf{X})$
- loi de probabilité stationnaire  $\pi$  à échantillonner

**Initialisation de la chaîne :** définir  $\mathbf{X}^0 = \mathbf{X}^{ini}$  et évaluer  $\pi(\mathbf{X}^0)$

**pour**  $n = 1$  à  $N + N_B$  **faire**

- Tirer une proposition  $\mathbf{X}^*$  de la loi de proposition  $q(\mathbf{X}^*|\mathbf{X})$  avec  $\mathbf{X} = \mathbf{X}^{n-1}$
- Évaluer  $\pi(\mathbf{X}^*)$ .
- Calculer le taux d'acceptation :

$$\alpha = \min\left(1, \frac{\pi(\mathbf{X}^*)}{\pi(\mathbf{X}^{n-1})}\right)$$

- Ajouter un n-ième élément de la chaîne  $\mathbf{X}^n = \mathbf{X}^*$  avec la probabilité  $\alpha$ , ou  $\mathbf{X}^n = \mathbf{X}^{n-1}$  avec la probabilité  $1 - \alpha$ .

**fin pour**

*Burn-in* : supprimer les  $N_B$  premiers échantillons de la chaîne.

**Sortie :** nuage de particules  $\{\mathbf{X}^n\}_{n=N_B+1, \dots, N_B+N}$  approximant la loi de probabilité  $\pi$

**2.4.2 Échantillonneur de Metropolis-Hastings**  $MH_D$ 

L'échantillonneur de Metropolis-Hastings est aussi une méthode itérative ou chaînée. Dans l'échantillonneur de Metropolis, le choix d'une loi de proposition  $q(\mathbf{X}^*|\mathbf{X})$  adaptative aux observations risque d'entrer en conflit avec la contrainte de parité de  $q(\mathbf{X}^*|\mathbf{X})$  imposée par l'algorithme 2.3. Hastings en a proposé une extension, levant ce conflit : l'algorithme 2.4 accepte une loi de proposition  $q(\mathbf{X}^*|\mathbf{X})$  quelconque. Il est important de noter que les propositions générées par les échantillonneur de Metropolis-Hastings (voir algorithme 2.4) proposent leurs nouveaux échantillons en effectuant des mouvements dans toutes les  $D$  dimensions de l'espace simultanément, ce qui est spécifié par l'indice  $D$ .

La progression du processus est illustrée sur la figure 2.8. La figure montre que les premières itérations sont fortement biaisées par la valeur initiale de la chaîne. Il est bénéfique de les éliminer de la chaîne finale : c'est le *Burn-in*. Le choix de la loi de proposition  $q(\mathbf{X}^*|\mathbf{X})$  est également déterminant pour mener une bonne exploration de l'espace. La figure 2.9 met l'algorithme en difficulté pour trouver le mode centré sur 60 avec une loi de proposition  $q(\mathbf{X}^*|\mathbf{X}) = \mathcal{N}(0, \sigma = 2)$ . Une variance de proposition plus élevée :  $q(\mathbf{X}^*|\mathbf{X}) = \mathcal{N}(0, \sigma = 10)$  permet de remédier à cette défaillance, comme illustré par la figure 2.10. Cette figure illustre par ailleurs la faculté de la méthode à estimer une loi de probabilité quelconque.

**2.4.3 Échantillonneur de Metropolis-Hastings à propositions marginalisées**  $MH_d$ 

L'échantillonneur de Metropolis-Hastings présenté plus haut utilise une fonction de proposition qui déplace l'échantillon dans toutes les dimensions simultanément. C'est pourquoi il peine à trouver les régions de forte probabilité lorsque la dimension de l'espace augmente. Cette difficulté est parfaitement expliquée dans (Smi07).

L'algorithme 2.5 est une variante de l'échantillonneur de Metropolis-Hastings, l'indice  $d$  signifiant que les nouveaux échantillons sont proposés en effectuant des mouvements marginalisés dans une seule des dimensions de l'espace. Dans cet algorithme, on partitionne le vecteur d'état  $\mathbf{X}^n$  de dimension  $D$  en  $S$  sous-vecteurs  $\mathbf{X}^n$ , tous de dimension  $d$ , tels que  $\mathbf{X}^n = [\mathbf{X}^{s,n}]_{s=1}^S$ , avec  $D = S.d$ . A chaque itération de la chaîne, on ne fait évoluer qu'un des vecteurs  $\mathbf{X}^n$ . Ceci revient à appliquer une succession d'itérations de Metropolis-Hastings dans des sous-espaces de dimensions  $d$  de l'espace d'état.

**Algorithme 2.4** Échantillonneur de Metropolis-Hastings  $MH_D$ **Entrées :**

- particule d'initialisation  $\mathbf{X}^{ini}$
- loi de probabilité de proposition  $q(\mathbf{X}^*|\mathbf{X})$
- loi de probabilité stationnaire  $\pi$  à échantillonner

**Initialisation de la chaîne :** définir  $\mathbf{X}^0 = \mathbf{X}^{ini}$  et évaluer  $\pi(\mathbf{X}^0)$

**pour**  $n = 1$  à  $N + N_B$  **faire**

- Tirer une proposition  $\mathbf{X}^*$  de la loi de proposition

$$q(\mathbf{X}^*|\mathbf{X}) \text{ avec } \mathbf{X} = \mathbf{X}^{n-1}$$

- Évaluer  $\pi(\mathbf{X}^*)$ .
- Calculer le taux d'acceptation :

$$\alpha = \min \left( 1, \frac{\pi(\mathbf{X}^*)Q(\mathbf{X}^{n-1}|\mathbf{X}^*)}{\pi(\mathbf{X}^{n-1})Q(\mathbf{X}^*|\mathbf{X}^{n-1})} \right)$$

- Ajouter un n-ième élément de la chaîne  $\mathbf{X}^n = \mathbf{X}^*$  avec la probabilité  $\alpha$ , ou  $\mathbf{X}^n = \mathbf{X}^{n-1}$  avec la probabilité  $1 - \alpha$ .

**fin pour**

*Burn-in* : supprimer les  $N_B$  premiers échantillons de la chaîne.

**Sortie :** nuage de particules  $\{\mathbf{X}^n\}_{n=N_B+1, \dots, N_B+N}$  approximant la loi de probabilité  $\pi$

**Algorithme 2.5** Échantillonneur de Metropolis-Hastings  $MH_d$ **Entrées :**

- particule d'initialisation  $\mathbf{X}^{ini} = [\mathbf{x}^{s,ini}]_{s=1}^S$ , avec  $\mathbf{X}^{ini}$  de dimension  $D$  et  $\mathbf{x}^{s,ini}$  de dimension  $d$ ,  $\forall s \in \{1, \dots, S\}$ , tels que  $D = S \cdot d$
- lois de probabilité de proposition marginales  $q_s(\mathbf{x}^*|\mathbf{x}^s)$ , applicables à  $\mathbf{x}^s$ ,  $s \in \{1, \dots, S\}$
- loi de probabilité stationnaire  $\pi$  à échantillonner

**Initialisation de la chaîne :** définir  $\mathbf{X}^0 = \mathbf{X}^{ini}$  et évaluer  $\pi(\mathbf{X}^0)$

**pour**  $n = 1$  à  $N + N_B$  **faire**

- Tirer l'indice  $s \in \{1, \dots, S\}$  du sous-vecteur à bouger
- Tirer une proposition  $\mathbf{x}^*$  de la loi de probabilité marginale

$$q_s(\mathbf{x}^*|\mathbf{x}) \text{ avec } \mathbf{x} = \mathbf{x}^{s,n-1}$$

- Générer  $\mathbf{X}^*$  en remplaçant  $\mathbf{x}^{s,n-1}$  par  $\mathbf{x}^*$  dans  $\mathbf{X}^{n-1}$ .
- Évaluer  $\pi(\mathbf{X}^*)$ .
- Calculer le taux d'acceptation :

$$\alpha = \min \left( 1, \frac{\pi(\mathbf{X}^*)Q_s(\mathbf{x}^{s,n-1}|\mathbf{x}^*)}{\pi(\mathbf{X}^{n-1})Q_s(\mathbf{x}^*|\mathbf{x}^{s,n-1})} \right)$$

- Ajouter un n-ième élément de la chaîne  $\mathbf{X}^n = \mathbf{X}^*$  avec la probabilité  $\alpha$ , ou  $\mathbf{X}^n = \mathbf{X}^{n-1}$  avec la probabilité  $1 - \alpha$ .

**fin pour**

*Burn-in* : supprimer les  $N_B$  premiers échantillons de la chaîne.

**Sortie :** nuage de particules  $\{\mathbf{X}^n\}_{n=N_B+1, \dots, N_B+N}$  approximant la loi de probabilité  $\pi$

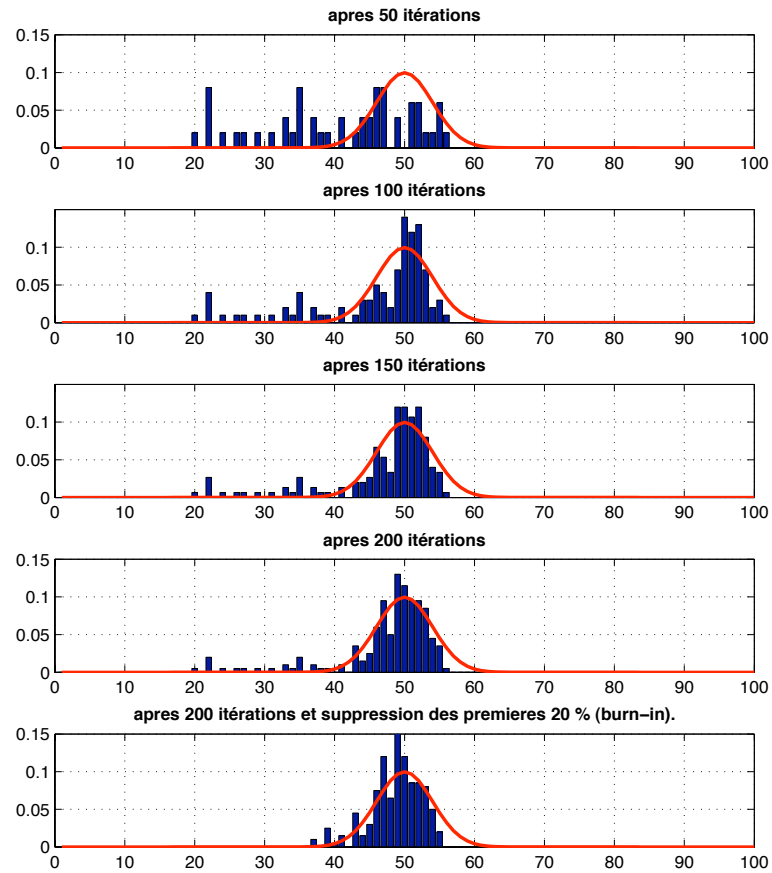


FIGURE 2.8 – Progression de l'échantillonnage par *MCMC* d'une loi de probabilité stationnaire monodimensionnelle unimodale  $\mathcal{N}(50, \sigma = 4)$ . Chaîne volontairement initialisée à  $\mathbf{X} = 20$  pour illustrer la nécessité du *Burn-in*. Loi de proposition  $q(\mathbf{X}^*|\mathbf{X}) = \mathcal{N}(0, \sigma = 2)$



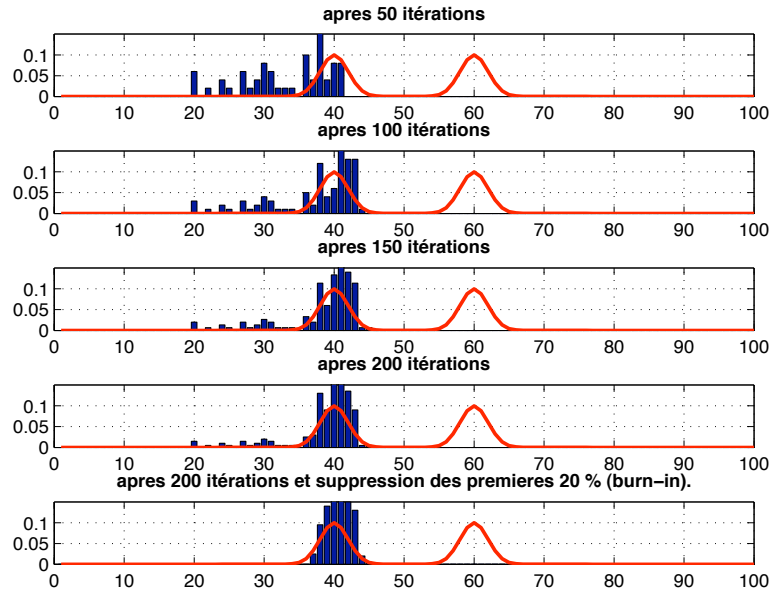


FIGURE 2.9 – Progression de l'échantillonnage par *MCMC* d'une loi de probabilité stationnaire monodimensionnelle bimodale  $0.5.(\mathcal{N}(40, \sigma = 2) + \mathcal{N}(60, \sigma = 2))$ . Chaîne volontairement initialisée à  $\mathbf{X} = 20$  pour illustrer la nécessité du *Burn-in*. Loi de proposition  $q(\mathbf{X}^*|\mathbf{X}) = \mathcal{N}(0, \sigma = 2)$

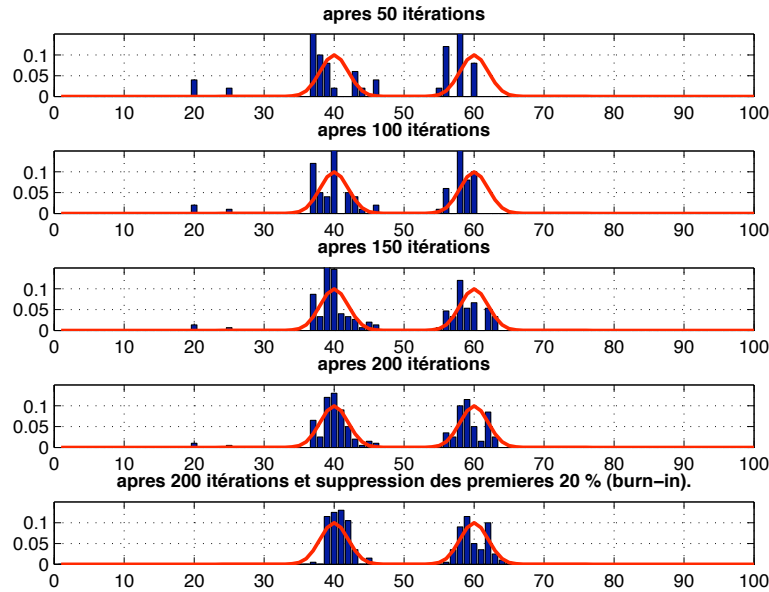


FIGURE 2.10 – Progression de l'échantillonnage par *MCMC* d'une loi de probabilité stationnaire monodimensionnelle bimodale  $0.5.(\mathcal{N}(40, \sigma = 2) + \mathcal{N}(60, \sigma = 2))$ . Chaîne volontairement initialisée à  $X = 20$  pour illustrer la nécessité du *Burn-in*. Loi de proposition  $q(\mathbf{X}^*|\mathbf{X}) = \mathcal{N}(0, \sigma = 10)$

## 2.5 Filtres Particulaires par *MCMC*

Également appelés Markov Chain Monte-Carlo Particle Filters (*MCMC PF*), ces filtres furent introduits par (KBD05) pour le suivi d'un grand nombre d'objets. Le principe proposé est de remplacer l'échantillonnage par importance (étape (a) de l'algorithme 2.1) par un échantillonnage de Metropolis. On a vu qu'un tel échantillonneur produit une approximation d'une loi de probabilité quelconque par un nuage d'échantillons non pondérés. Le Filtre Particulaire *MCMC* applique un échantillonnage de Metropolis pour approximer  $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$ , loi de probabilité *a posteriori* de l'état à chaque pas temporel  $t$ . Dans la section 2.5.1, nous présentons un Filtre Particulaire *MCMC* dont les nouveaux échantillons proposés sont générés en effectuant des mouvements dans toutes les  $D$  dimensions de l'espace simultanément. Nous le notons *FP MCMC<sub>D</sub>*. On examinera dans la section 2.5.2 le *FP MCMC<sub>d</sub>* une variante du *FP MCMC<sub>D</sub>*, dans lequel chaque nouvel échantillon proposé est généré en ne perturbant que  $d$  dimensions de l'état de l'échantillon précédent, avec  $0 \leq d \leq D$ , c'est à dire que les propositions se font sur un sous-espace de dimension  $d$ . Supposons que l'on dispose d'une approximation de  $p(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1})$  par  $N$  échantillons discrets non pondérés  $\{\mathbf{X}_{t-1}^\nu\}_{\nu=1}^N$  :

$$p(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1}) \approx \frac{1}{N} \sum_{\nu=1}^N \delta(\mathbf{X}_{t-1} - \mathbf{X}_{t-1}^\nu). \quad (2.7)$$

L'équation (2.8) délivre  $p(\mathbf{X}_t|\mathbf{Z}_{1:t-1})$ , loi de probabilité *a priori* de l'état à l'instant  $t$ , approximation de l'équation (1.6) par  $N$  composantes de prédiction dynamique non pondérées :

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t-1}) \approx \frac{1}{N} \sum_{\nu=1}^N p(\mathbf{X}_t|\mathbf{X}_{t-1}^\nu), \quad (2.8)$$

où  $p(\mathbf{X}_t|\mathbf{X}_{t-1})$  est la loi de probabilité d'évolution dynamique du système. La loi définie par l'équation (2.8) est la moyenne des  $N$  composantes  $p(\mathbf{X}_t|\mathbf{X}_{t-1}^\nu)$ . L'approximation échantillonnée de l'équation (1.7) du filtre Bayésien récursif est toujours donnée par l'équation (2.5). Seule la constante  $C$  est modifiée, ce qui est sans influence, puisque la loi est normalisée.

### 2.5.1 Filtre Particulaire *MCMC<sub>D</sub>*

Dans le Filtre Particulaire *MCMC<sub>D</sub>*, les nouveaux échantillons proposés sont générés en perturbant simultanément toutes les  $D$  dimensions de l'espace. De même que dans l'algorithme 2.1, on échantillonne en choisissant pour loi de proposition  $q$ , l'approximation échantillonnée de la loi *a priori* :

$$\mathbf{X}_t^n \sim q(\mathbf{X}_t) = \frac{1}{N} \sum_{n=1}^N p(\mathbf{X}_t|\mathbf{X}_{t-1}^n) \quad (2.9)$$

La différence est que le rééchantillonnage de la loi de probabilité *a posteriori* réalisé à chaque pas temporel, est mené par un échantillonneur de Metropolis-Hastings. Comme ce dernier, le filtre approxime alors  $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$ , loi de probabilité *a posteriori* de l'état à l'instant  $t$ , par le lot de  $N$  échantillons discrets non pondérés  $\{\mathbf{X}_t^n\}_{n=1}^N$ . La figure 2.11 illustre une itération du Filtre Particulaire *MCMC*. L'algorithme 2.6 résume le fonctionnement de ce filtre intégrant un échantillonneur de Metropolis-Hastings : l'exploration est menée par une chaîne de Markov selon l'algorithme 2.4, où la loi cible  $\pi$  est remplacée par  $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$ , loi de probabilité *a posteriori* de l'état à chaque pas temporel  $t$ . La loi de proposition  $q$  est définie par (2.9), mixture des lois de prédiction dynamique appliquées au nuage de particules approximant la loi *a posteriori* à l'instant précédent. Le taux d'acceptation à l'itération  $n$  est alors :

$$\alpha = \min \left( 1, \frac{P(\mathbf{X}_t^*|\mathbf{Z}_{1:t})Q(\mathbf{X}_t^{n-1})}{P(\mathbf{X}_t^{n-1}|\mathbf{Z}_{1:t})Q(\mathbf{X}_t^*)} \right). \quad (2.10)$$

L'application de la règle de Bayes à  $P(\mathbf{X}_t^*|\mathbf{Z}_{1:t})$  et à  $P(\mathbf{X}_t^{n-1}|\mathbf{Z}_{1:t})$  permet de développer  $\alpha$  :

$$\alpha = \min \left( 1, \frac{P(\mathbf{Z}_t|\mathbf{X}_t^*)P(\mathbf{X}_t^*|\mathbf{Z}_{1:t-1})Q(\mathbf{X}_t^{n-1})}{P(\mathbf{Z}_t|\mathbf{X}_t^{n-1})P(\mathbf{X}_t^{n-1}|\mathbf{Z}_{1:t-1})Q(\mathbf{X}_t^*)} \right) \quad (2.11)$$

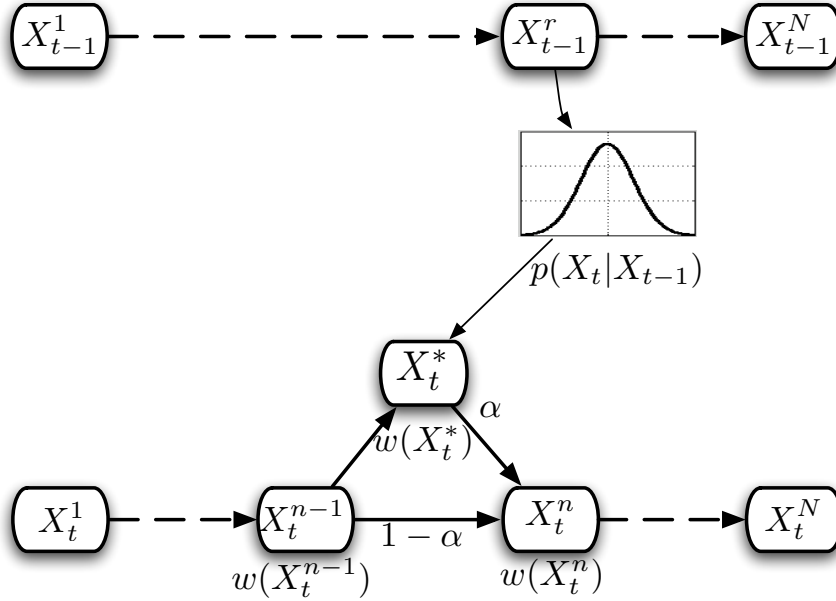


FIGURE 2.11 – Une itération du *Filtre Particulaire MCMC* : une particule  $\mathbf{X}_{t-1}^r$  est tirée aléatoirement de la chaîne à l’instant précédent. Une proposition  $\mathbf{X}_t^*$  est alors générée à partir de la loi  $q(\mathbf{X}_t^* | \mathbf{X}_{t-1}^r)$ . Elle est acceptée avec la probabilité  $\alpha$  selon l’équation 2.12, sinon l’état précédent  $\mathbf{X}_t^{n-1}$  est dupliqué.

Or  $q(\mathbf{X}_t)$  défini par l’équation (2.9) n’est autre que l’approximation échantillonnée de  $p(\mathbf{X}_t | \mathbf{Z}_{1:t-1})$  défini par l’équation (2.8), donc  $P(\mathbf{X}_t^* | \mathbf{Z}_{1:t-1}) \approx Q(\mathbf{X}_t^*)$  et  $P(\mathbf{X}_t^{n-1} | \mathbf{Z}_{1:t-1}) \approx Q(\mathbf{X}_t^{n-1})$ , ce qui simplifie considérablement les calculs :

$$\alpha = \min \left( 1, \frac{P(\mathbf{Z}_t | \mathbf{X}_t^*)}{P(\mathbf{Z}_t | \mathbf{X}_t^{n-1})} \right) \quad (2.12)$$

Cet algorithme présente deux différences par rapport à l’algorithme *SIR* (algorithme 2.1) : 1) La constitution du nuage de particules est séquentielle. 2) L’approximation des lois *a posteriori* se fait ici par nuage de particules non pondérées, alors qu’elles sont pondérées dans le *SIR*. Tel quel, cet algorithme peut être vu comme une réécriture chaînée du Filtre Particulaire parallèle, algorithme *SIR* (algorithme 2.1), et où les poids des particules ont été remplacés par des densités de population de l’espace par les particules. Il présente donc la même faiblesse : il peinera à échantillonner convenablement une loi de probabilité dans un espace d’état de grande dimension.

### 2.5.2 Filtre Particulaire $MCMC_d$

On verra que le Filtre Particulaire  $MCMC_D$  est performant dans les espaces de petites dimensions, mais est inefficace dans les espaces de grande dimension, car l’étape de prédiction mobilise à chaque itération la totalité des composantes de  $\mathbf{X}_t^n$ . Dans le cas des échantillonneurs, on a rencontré précédemment cette difficulté avec l’algorithme 2.4, et on a présenté avec l’algorithme 2.5 une stratégie permettant de la contourner, en effectuant des mouvements sur un sous-espace et non sur la totalité de l’espace d’état. Cet algorithme peut être utilisé pour réaliser le rééchantillonnage des Filtres Particulaires  $MCMC$  : c’est l’intérêt du filtre présenté par Khan *et al.* (KBD04; KBD05). Le  $FP\ MCMC_d$  est une extension du  $FP\ MCMC_D$ , dans lequel on partitionne le vecteur d’état  $\mathbf{X}_t^n$  de dimension  $D$  en  $S$  sous-vecteurs  $\mathbf{x}_t^{s,n}$ , tous de dimension  $d$ , tels que  $\mathbf{X}_t^n = [\mathbf{x}_t^{s,n}]_{s=1}^S$ , avec  $D = S.d$ . A chaque itération de la chaîne, le  $FP\ MCMC_d$  tire l’indice  $s$  du sous-espace où va se produire le mouvement. Notons  $p(\mathbf{x}_t^s | \mathbf{x}_{t-1}^s)$  la loi de probabilité modélisant la dynamique marginalisée sur le sous-espace

**Algorithme 2.6** Filtre Particulaire *MCMC<sub>D</sub>***Entrées :**

- nuage de particules approximant la loi de probabilité *a posteriori* de l'état  $\mathbf{X}_{t-1}$  à l'instant  $t - 1$  :  $\{\mathbf{X}_{t-1}^n\}_{n=1}^N$
- loi de probabilité de prédiction  $p(\mathbf{X}_t|\mathbf{X}_{t-1})$  modélisant la dynamique de  $\mathbf{X}$ .
- modèle de vraisemblance d'observation  $p(\mathbf{Z}|\mathbf{X})$ .
- observation  $\mathbf{Z}_t$  à l'instant  $t$ .

**Initialisation de la chaîne :**

- Tirer aléatoirement une particule de la chaîne précédente :  $\mathbf{X}_{t-1}^r, r \in \{1, \dots, N\}$ .
- Générer une particule initiale  $\mathbf{X}_t^0$  tirée de la loi de prédiction :  $p(\mathbf{X}_t|\mathbf{X}_{t-1})$  avec  $\mathbf{X}_{t-1} = \mathbf{X}_{t-1}^r$
- Évaluer son poids  $w(\mathbf{X}_t^0) \propto P(\mathbf{Z}_t|\mathbf{X}_t^0)$ , vraisemblance de l'observation  $\mathbf{Z}_t$  sachant l'état  $\mathbf{X}_t^0$ .

**pour**  $i = 1$  à  $N + N_B$  **faire**

- Propagation : tirer aléatoirement une particule de la chaîne précédente :  $\mathbf{X}_{t-1}^r, r \in \{1, \dots, N\}$
- Prédiction : générer une particule prédite  $\mathbf{X}_t^*$  tirée de la loi de prédiction :  $p(\mathbf{X}_t|\mathbf{X}_{t-1})$  avec  $\mathbf{X}_{t-1} = \mathbf{X}_{t-1}^r$
- Évaluation : calculer  $w(\mathbf{X}_t^*) \propto P(\mathbf{Z}_t|\mathbf{X}_t^*)$ , vraisemblance de l'observation  $\mathbf{Z}_t$  sachant l'état  $\mathbf{X}_t^*$ .
- Calcul du taux d'acceptation :

$$\alpha = \min \left( 1, \frac{w(\mathbf{X}_t^*)}{w(\mathbf{X}_t^{i-1})} \right)$$

- Ajout du  $i^{\text{ième}}$  élément de la chaîne :  $\mathbf{X}_t^i = \mathbf{X}_t^*$  avec la probabilité  $\alpha$ , ou  $\mathbf{X}_t^i = \mathbf{X}_t^{i-1}$  avec la probabilité  $1 - \alpha$ .

**fin pour**

**Burn-in** : supprimer les  $N_B$  premiers échantillons de la chaîne.

**Sortie** : nuage de particules approximant la loi de probabilité *a posteriori* de  $\mathbf{X}_t$  à l'instant  $t$  :

$$\{\mathbf{X}_t^n\}_{n=N_B+1, \dots, N_B+N}$$

$s$ , que nous allons utiliser pour expliciter les facteurs nécessaires au calcul du taux d'acceptation défini en 2.11.

$$p(\mathbf{X}_t | \mathbf{X}_{t-1}) = \prod_{s=1}^S p(\mathbf{x}_t^s | \mathbf{x}_{t-1}^s), \quad (2.13)$$

que l'on utilise pour réécrire l'approximation échantillonnée de la loi *a priori* (2.8) :

$$p(\mathbf{X}_t | \mathbf{Z}_{1:t-1}) \approx \frac{1}{N} \sum_{\nu=1}^N p(\mathbf{X}_t | \mathbf{X}_{t-1}^\nu) = \frac{1}{N} \sum_{\nu=1}^N \prod_{s=1}^S p(\mathbf{x}_t^s | \mathbf{x}_{t-1}^{s,\nu}) \quad (2.14)$$

A chaque itération  $n$ , on propose un nouvel état joint  $\mathbf{X}_t^*$ , généré par mouvement marginalisé sur un des sous-espaces  $s \in \{1, \dots, S\}$ . Notons  $s^*$  ce sous-espace. Pour calculer le taux d'acceptation, il nous faut évaluer cette loi pour l'état proposé  $\mathbf{X}_t^*$  et pour l'état précédent  $\mathbf{X}_t^{n-1}$  :

$$P(\mathbf{X}_t^* | \mathbf{Z}_{1:t-1}) \approx \frac{1}{N} \sum_{\nu=1}^N P(\mathbf{x}_t^* | \mathbf{x}_{t-1}^{s^*,\nu}) \prod_{s=1, s \neq s^*}^S P(\mathbf{x}_t^s | \mathbf{x}_{t-1}^s) \quad (2.15)$$

$$P(\mathbf{X}_t^{n-1} | \mathbf{Z}_{1:t-1}) \approx \frac{1}{N} \sum_{\nu=1}^N P(\mathbf{x}_t^{n-1} | \mathbf{x}_{t-1}^{s^*,\nu}) \prod_{s=1, s \neq s^*}^S P(\mathbf{x}_t^{n-1} | \mathbf{x}_{t-1}^s) \quad (2.16)$$

Comme les projections de  $\mathbf{X}_t^*$  et  $\mathbf{X}_t^{n-1}$  sur tout sous-espace  $s \neq s^*$  sont confondues, on peut noter :

$$a_t^\nu = \prod_{s=1, s \neq s^*}^S P(\mathbf{x}_t^* | \mathbf{x}_{t-1}^{s,\nu}) = \prod_{s=1, s \neq s^*}^S P(\mathbf{x}_t^{n-1} | \mathbf{x}_{t-1}^{s,\nu}), \forall \nu \in \{1, \dots, N\} \quad (2.17)$$

D'autre part, les configurations jointes  $\mathbf{X}_t^*$  sont tirées selon la loi de proposition :

$$\mathbf{X}_t^* \sim q(\mathbf{X}_t | \mathbf{X}_t^{n-1}) = \sum_{s=1}^S q(s) q(\mathbf{X}_t | \mathbf{X}_t^{n-1}, s), \quad (2.18)$$

où  $q(s)$  est la loi de probabilité conditionnant le choix du sous-espace où se fait le mouvement, tandis que  $q(\mathbf{X}_t | \mathbf{X}_t^{n-1}, s)$  est la loi de proposition de  $\mathbf{X}_t$ , conditionnée par le choix du sous-espace  $s$  :

$$q(\mathbf{X}_t | \mathbf{X}_t^{n-1}, s) = \frac{1}{N} \sum_{\nu=1}^N p(\mathbf{x}_t^s | \mathbf{x}_{t-1}^{s,\nu}) \prod_{s=1, s \neq s^*}^S \delta(\mathbf{x}_t^s - \mathbf{x}_t^{s,n-1}) \quad (2.19)$$

L'équation (2.18) peut alors se réécrire :

$$q(\mathbf{X}_t | \mathbf{X}_t^{n-1}) = \sum_{s=1}^S q(s) \frac{1}{N} \sum_{\nu=1}^N p(\mathbf{x}_t^s | \mathbf{x}_{t-1}^{s,\nu}) \prod_{s=1, s \neq s^*}^S \delta(\mathbf{x}_t^s - \mathbf{x}_t^{s,n-1}) \quad (2.20)$$

L'évaluation de la loi (2.20) pour l'état proposé  $\mathbf{X}_t^*$  et pour l'état initial  $\mathbf{X}_t^{n-1}$ , donne respectivement :

$$Q(\mathbf{X}_t^* | \mathbf{X}_t^{n-1}) = Q(s^*) \frac{1}{N} \sum_{\nu=1}^N P(\mathbf{x}_t^* | \mathbf{x}_{t-1}^{s^*,\nu}) \quad (2.21)$$

$$Q(\mathbf{X}_t^{n-1} | \mathbf{X}_t^*) = Q(s^*) \frac{1}{N} \sum_{\nu=1}^N P(\mathbf{x}_t^{n-1} | \mathbf{x}_{t-1}^{s^*,\nu}) \quad (2.22)$$

Compte tenu des équations (2.15), (2.16), (2.17) (2.21), et (2.22), le taux d'acceptation (2.11) devient alors :

$$\alpha = \min \left( 1, \frac{P(\mathbf{Z}_t | \mathbf{X}_t^*) \left( \sum_{\nu=1}^N a_t^\nu P(\mathbf{x}_t^* | \mathbf{x}_{t-1}^{s^*,\nu}) \right) \left( \sum_{\nu=1}^N P(\mathbf{x}_t^{n-1} | \mathbf{x}_{t-1}^{s^*,\nu}) \right)}{P(\mathbf{Z}_t | \mathbf{X}_t^{n-1}) \left( \sum_{\nu=1}^N a_t^\nu P(\mathbf{x}_t^{n-1} | \mathbf{x}_{t-1}^{s^*,\nu}) \right) \left( \sum_{\nu=1}^N P(\mathbf{x}_t^* | \mathbf{x}_{t-1}^{s^*,\nu}) \right)} \right) \quad (2.23)$$

Cette équation montre malheureusement que la simplification qui nous avait conduit à l'équation 2.12 dans le cas du filtre non marginalisé noté *FP MCMC<sub>D</sub>*, n'est plus possible ici. Le calcul rigoureux du taux d'acceptation requiert donc les lourds calculs de l'équation 2.23 à chaque itération. Notre priorité étant le temps réel, et compte tenu des puissances de calcul actuellement disponibles, nous avons choisi pour le code implémenté en octobre 2009, de ne calculer que la version simplifiée du taux d'acceptation, conformément à 2.12 :

$$\alpha = \min \left( 1, \frac{P(\mathbf{Z}_t | \mathbf{X}_t^*)}{P(\mathbf{Z}_t | \mathbf{X}_t^{n-1})} \right)$$

La marginalisation de l'espace peut être choisie en fonction du besoin : dans le cas du suivi multi-objet, l'approche la plus fréquente consiste à affecter un sous-espace à la description de l'état de chaque objet. Ce point est discuté dans le chapitre 3. La méthode, exposée dans l'algorithme 2.7, conserve alors ses performances, car on marginalise ainsi les propositions de mouvements des composantes de l'état dans des sous espaces de dimension modeste. Cette opération peut être vue comme une proposition de croisement génétique entre la particule courante et une prédiction générée à partir d'une des particules de la chaîne précédente. L'application typique de cet algorithme au suivi d'objet, consiste à choisir des vecteurs  $\mathbf{x}_t^s$  décrivant chacun un objet,  $S$  est alors le nombre d'objets suivis. Pour d'autres applications, où les composantes sont indépendamment observables, on peut pousser la logique jusqu'au *FP MCMC<sub>1</sub>*, l'indice 1 signifiant que les nouveaux échantillons sont proposés en effectuant des mouvements marginaux dans une seule dimension à la fois.

---

**Algorithme 2.7** Filtre Particulaire *MCMC<sub>d</sub>*


---

**Entrées :**

- nuage de particules approximant la loi de probabilité *a posteriori* de l'état  $\mathbf{X}_{t-1}$  à l'instant  $t - 1$  :  $\{\mathbf{X}_{t-1}^n\}_{n=1}^N$ , tel que  $\mathbf{X}_{t-1}^n = [\mathbf{x}_{t-1}^{s,n}]_{s=1}^S$ , avec  $\mathbf{X}_{t-1}^n$  de dimension  $D$  et  $\mathbf{x}_{t-1}^{s,n}$  de dimension  $d$ ,  $\forall s \in \{1, \dots, S\}$ , tels que  $D = S.d$
- lois de probabilité de prédiction marginales  $p_s(\mathbf{x}_t^s | \mathbf{x}_{t-1}^s)$ , modélisant la dynamique de  $\mathbf{x}^s$ ,  $s \in \{1, \dots, S\}$
- modèle de vraisemblance d'observation  $p(\mathbf{Z} | \mathbf{X})$ .
- observation  $\mathbf{Z}_t$  à l'instant  $t$ .

**Initialisation de la chaîne :**

- Tirer aléatoirement une particule de la chaîne précédente :  $\mathbf{X}_{t-1}^r$ ,  $r \in \{1, \dots, N\}$ .
- Générer une particule initiale  $\mathbf{X}_t^0 = [\mathbf{x}_t^{s,0}]_{s=1}^S$ , où les  $\mathbf{x}_t^{s,0} \forall s \in \{1, \dots, S\}$  sont tirées des lois de prédiction marginales :  $p_s(\mathbf{x}_t^s | \mathbf{x}_{t-1}^{s,r})$ .
- Évaluer son poids  $w(\mathbf{X}_t^0) \propto P(\mathbf{Z}_t | \mathbf{X}_t^0)$ , vraisemblance de l'observation  $\mathbf{Z}_t$  sachant l'état  $\mathbf{X}_t^0$ .

**pour**  $i = 1$  à  $N + N_B$  **faire**

- Tirer l'indice du sous-vecteur à bouger :  $s \in \{1, \dots, S\}$
- Propagation : tirer aléatoirement une particule de la chaîne précédente :  $\mathbf{X}_{t-1}^r$ ,  $r \in \{1, \dots, N\}$ , et extraire le vecteur :  $\mathbf{x}_{t-1}^{s,r}$
- Prédiction : générer  $\mathbf{x}^*$ , tirée de la loi marginale dynamique  $p_s(\mathbf{x}_t^s | \mathbf{x}_{t-1}^{s,r})$ .
- Générer une particule prédite  $\mathbf{X}^*$ , en remplaçant le vecteur  $\mathbf{x}_t^{s,i-1}$  de  $\mathbf{X}_t^{i-1}$ , par  $\mathbf{x}^*$ .
- Évaluation : calculer  $w(\mathbf{X}^*) \propto P(\mathbf{Z}_t | \mathbf{X}^*)$ , vraisemblance de l'observation  $\mathbf{Z}_t$  sachant l'état  $\mathbf{X}^*$ .
- Calcul du taux d'acceptation selon l'équation (2.23).
- Ajout du  $i^{\text{ème}}$  élément de la chaîne :  $\mathbf{X}_t^i = \mathbf{X}^*$  avec la probabilité  $\alpha$ , ou  $\mathbf{X}_t^i = \mathbf{X}_t^{i-1}$  avec la probabilité  $1 - \alpha$ .

**fin pour**

**Burn-in** : supprimer les  $N_B$  premiers échantillons de la chaîne.

**Sortie** : nuage de particules approximant la loi de probabilité *a posteriori* de  $\mathbf{X}_t$  à l'instant  $t$  :  $\{\mathbf{X}_t^n\}_{n=N_B+1, \dots, N_B+N}$

---

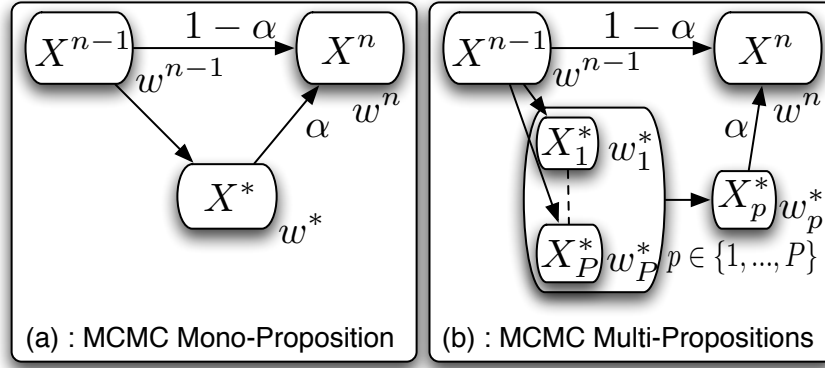


FIGURE 2.12 – (a) : la  $n^{\text{ième}}$  itération du  $FP\ MCMC^1$  produit une seule proposition  $\mathbf{X}^*$ , acceptée comme prochain maillon  $\mathbf{X}^n$  avec la probabilité  $\alpha$ . (b) : la  $n^{\text{ième}}$  itération du  $FP\ MCMC^P$  produit  $P$  propositions  $\mathbf{X}_p^*$ . L'une d'elles est tirée avec une probabilité égale à son poids normalisé  $\hat{w}_p^* = \frac{w_p^*}{\sum_{p=1}^P w_p^*}$ , puis acceptée comme prochain maillon  $\mathbf{X}^n$  avec la probabilité  $\alpha$ .

## 2.6 Filtre Particulaire $MCMC$ Parallélisé

Cette section présente une des contributions de cette thèse : une approximation du Filtre Particulaire  $MCMC$ , permettant d'exploiter le potentiel de calcul des machines parallèles. Depuis quelques années, l'industrie de l'informatique développe des unités à plusieurs processeurs ou des processeurs à plusieurs cœurs. Pour bénéficier de cette puissance parallèle, il faut équilibrer les charges de calculs entre les cœurs. Dans les différentes versions de Filtres Particulaires  $MCMC$  examinées jusqu'ici, le calcul de la vraisemblance de la mesure à chaque proposition est de loin le plus coûteux, car il implique toute l'image (ces calculs seront détaillées dans la section 4.4). Il est ainsi le meilleur candidat à la parallélisation. Malheureusement, cette tâche est nécessaire à chaque itération de la chaîne, et son résultat est utilisé pour décider si la proposition devrait être acceptée ou non comme nouvelle particule de la chaîne. Nous noterons désormais ces filtres  $FP\ MCMC^1$ , où l'exposant <sup>1</sup> désigne le nombre de propositions générées à chaque itération de la chaîne. Un  $FP\ MCMC^1$  ne peut pas être parallélisé tel quel. Pour bénéficier de cette puissance parallèle, nous proposons une approximation du  $FP\ MCMC^1$ , qui génère et évalue en parallèle plusieurs propositions. Il est noté  $FP\ MCMC^P$ , où l'exposant <sup>P</sup> désigne le nombre d'unités de traitement de la machine. L'objectif est de réduire le nombre de particules. Omettant le temps par souci de simplicité, la figure 2.12-b illustre une itération du  $FP\ MCMC^P$ , transition markovienne de la particule  $\mathbf{X}^{n-1}$  à la particule  $\mathbf{X}^n$  via  $P$  nouvelles propositions  $\mathbf{X}_p^*$ ,  $p \in \{1, \dots, P\}$ . Chaque cœur génère une proposition d'état, évalue sa vraisemblance  $w_p^* \propto P(\mathbf{Z}_t | \mathbf{X}_p^*)$  et la retourne au processus principal. L'une d'elles est alors tirée parmi toutes les propositions parallèles, selon leurs vraisemblances respectives  $w_p^*$ . Elle est acceptée comme nouveau maillon  $\mathbf{X}^n$  avec la probabilité  $\alpha$  définie dans l'algorithme 2.8. Si elle est refusée alors  $\mathbf{X}^n = \mathbf{X}^{n-1}$ . L'algorithme 2.8 est ainsi une approximation de l'algorithme 2.7, permettant de proposer et évaluer  $P$  hypothèses en parallèle à chaque itération, au lieu d'une seule. La figure 2.13 montre qualitativement que les échantillonneurs  $MCMC$  mono et multi-proposition ont des comportements similaires par rapport à la densité cible. La figure 2.14 montre que notre  $FP\ MCMC^P$  se comporte de la manière similaire quelle que soit la valeur de  $P$ . Les particules sont diffusées dynamiquement à partir de  $t = 1$  vers les nouvelles loi de probabilité. Elle montre également que le filtre  $FP\ MCMC^P$  reste capable d'approximer une loi multi-modale. Elle montre enfin que les filtres à propositions multiples convergent plus rapidement, permettant de réduire la longueur de la chaîne. Nous ne pensons pas que le  $FP\ MCMC^P$  réalise une approximation convergeant vers la loi de probabilité *a posteriori* comme le fait le  $FP\ MCMC^1$  lorsqu'on augmente le nombre de particules. Nous n'avons pas d'éléments tangibles pour répondre à cette question, et des investigations complémentaires seraient nécessaires. Nous l'utiliserons néanmoins dans les expérimentations, dans le but de respecter les contraintes de temps réel.



**Algorithme 2.8** Filtre Particulaire *MCMC*<sub>d</sub><sup>p</sup>**Entrées :**

- nuage de particules approximant la loi de probabilité *a posteriori* de l'état  $\mathbf{X}_{t-1}$  à l'instant  $t - 1$  :  $\{\mathbf{X}_{t-1}^n\}_{n=1}^N$ , tel que  $\mathbf{X}_{t-1}^n = [\mathbf{X}_{t-1}^{s,n}]_{s=1}^S$ , avec  $\mathbf{X}_{t-1}^n$  de dimension  $D$  et  $\mathbf{X}_{t-1}^{s,n}$  de dimension  $d$ ,  $\forall s \in \{1, \dots, S\}$ , tels que  $D = S.d$
- lois de probabilité de prédiction marginales  $p_s(\mathbf{x}_t^s | \mathbf{x}_{t-1}^s)$ , modélisant la dynamique de  $\mathbf{x}^s$ ,  $s \in \{1, \dots, S\}$
- modèle de vraisemblance d'observation  $p(\mathbf{Z} | \mathbf{X})$ .
- observation  $\mathbf{Z}_t$  à l'instant  $t$ .

**Initialisation de la chaîne :**

- Tirer aléatoirement une particule de la chaîne précédente :  $\mathbf{X}_{t-1}^r$ ,  $r \in \{1, \dots, N\}$ .
- Générer une particule initiale  $\mathbf{X}_t^0 = [\mathbf{x}_t^{s,0}]_{s=1}^S$ , où les  $\mathbf{x}_t^{s,0} \forall s \in \{1, \dots, S\}$  sont tirés des lois de prédiction marginales :  $p_s(\mathbf{x}_t^s | \mathbf{x}_{t-1}^{s,r})$ .
- Évaluer son poids  $w(\mathbf{X}_t^0) \propto P(\mathbf{Z}_t | \mathbf{X}_t^0)$ , vraisemblance de l'observation  $\mathbf{Z}_t$  sachant l'état  $\mathbf{X}_t^0$ .

**pour**  $i = 1$  à  $N + N_B$  **faire****pour**  $p = 1$  à  $P$  **faire**

- Tirer l'indice du sous-vecteur à bouger :  $s_p \in \{1, \dots, S\}$
- Propagation : tirer aléatoirement une particule de la chaîne précédente :  $\mathbf{X}_{t-1}^{r_p}$ ,  $r_p \in \{1, \dots, N\}$ , et extraire le vecteur :  $\mathbf{x}_{t-1}^{s_p, r_p}$
- Prédiction : générer  $\mathbf{x}_{t,p}^*$ , tirée de la loi marginale de prédiction  $p_{s_p}(\mathbf{x}_{t,p}^* | \mathbf{x}_{t-1}^{s_p, r_p})$ .
- Générer une particule prédite  $\mathbf{X}_p^*$ , en remplaçant le vecteur  $\mathbf{x}_{t-1}^{s_p, i-1}$  par  $\mathbf{x}_{t,p}^*$  dans  $\mathbf{X}_{t-1}^{i-1}$ .
- Évaluation : calculer  $w_p^* \propto P(\mathbf{Z}_t | \mathbf{X}_p^*)$ , vraisemblance de l'observation  $\mathbf{Z}_t$  sachant l'état  $\mathbf{X}_p^*$ .

**fin pour**

- Calculer les poids normalisés  $\hat{w}_p^* = \frac{w_p^*}{\sum_{p=1}^P w_p^*}$
- Tirer  $\{\mathbf{X}^*, w^*\} = \{\mathbf{X}_{p_0}^*, w_{p_0}^*\}$ , avec la probabilité  $\hat{w}_{p_0}^*$  parmi ces  $P$  prédictions pondérées :  $\{\mathbf{X}_p^*, w_p^*\}_{p=1}^P$ .
- Calcul du taux d'acceptation selon l'équation (2.23).
- Ajout du  $i^{\text{ème}}$  élément de la chaîne :  $\mathbf{X}_t^i = \mathbf{X}_{p_0}^*$  avec la probabilité  $\alpha$ , ou  $\mathbf{X}_t^i = \mathbf{X}_t^{i-1}$  avec la probabilité  $1 - \alpha$ .

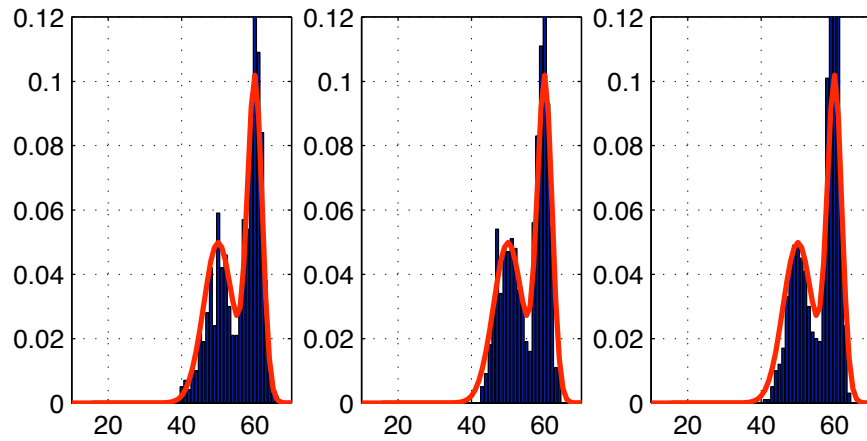
**fin pour****Burn-in** : supprimer les  $N_B$  premiers échantillons de la chaîne.**Sortie** : nuage de particules approximant la loi de probabilité *a posteriori* de  $\mathbf{X}_t$  à l'instant  $t$  : $\{\mathbf{X}_t^n\}_{n=N_B+1, \dots, N_B+N}$ 

FIGURE 2.13 – Échantillonnage d'une densité monodimensionnelle (courbe continue), par un *MCMC* (histogramme). À gauche : simple proposition, au centre : double proposition, à droite : 4 propositions.  $N = 1000$  itérations, avec loi de proposition  $\mathcal{N}(x^{n-1}, 5^2)$ .

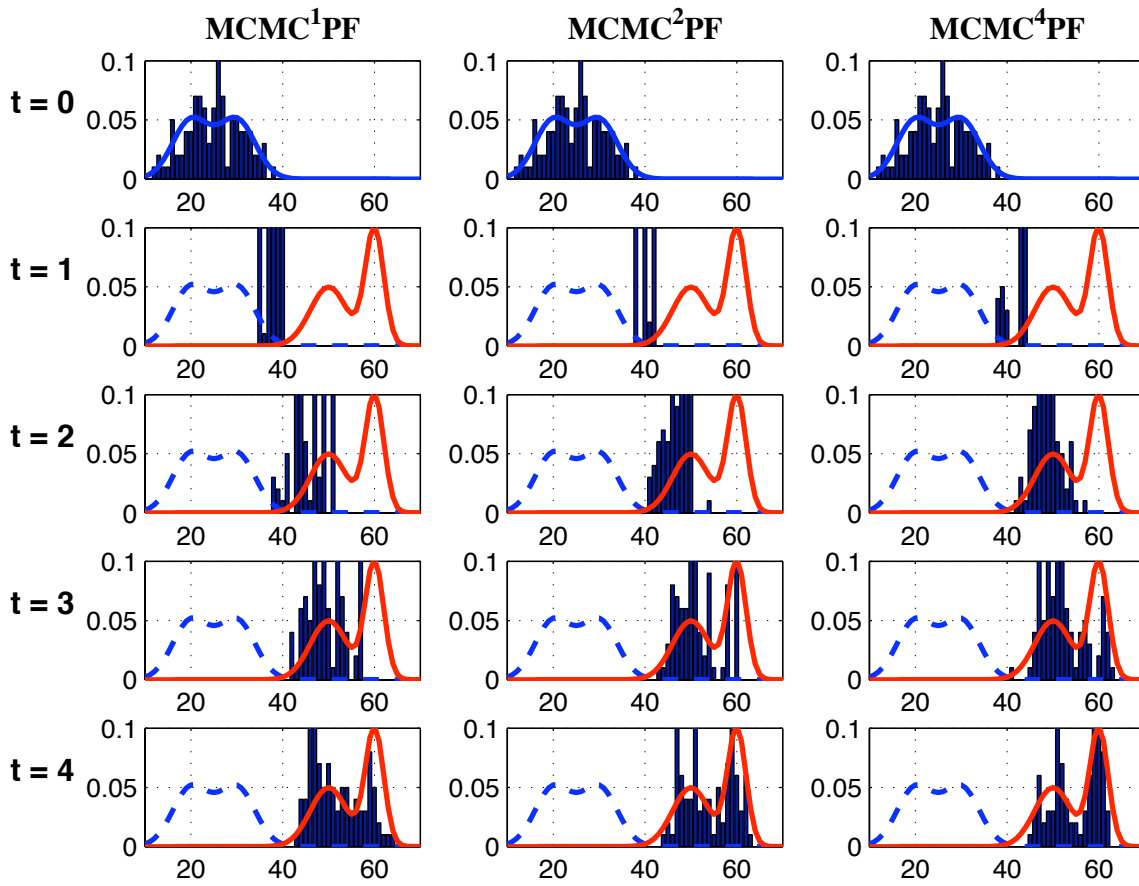


FIGURE 2.14 – Filtrage d’une densité pour plusieurs configurations du  $FP\ MCMC^P$ . En pointillés, la densité de probabilité *a posteriori* initiale à  $t=0$ . En trait continu, la fonction de vraisemblance stationnaire de l’instant  $t=1$  (2e ligne) jusqu’à  $t=4$  (bas). La loi de probabilité des particules est tracée sous forme d’histogramme. À gauche : simple proposition  $P = 1$ , colonne centrale : double proposition  $P = 2$ , à droite : 4 propositions  $P = 4$ .  $N = 100$  particules avec une loi de proposition :  $\mathcal{N}(x_{t-1}^{n-1}, 5^2)$ .

## 2.7 Conclusion

Dans ce chapitre, nous avons distingué deux familles de filtres particulières très utilisés dans les applications de suivi Multi-Objets par vision artificielle : parallèles ou itératives. Dans la famille des méthodes parallèles, on trouve les Filtres Particulaires *SIR* et les Filtres Particulaires *SIR Partitionnés*. L'avantage de ces méthodes est de pouvoir être aisément implémentées sur ordinateurs parallèles. Dans la famille des méthodes chaînées ou itératives, on trouve les Filtres Particulaires *MCMC*, que l'on a déclinés en deux variantes : *MCMC<sub>D</sub>* et *MCMC<sub>d</sub>*. Ces méthodes, du fait de leur structure itérative, ne peuvent être implémentées efficacement telles quelles sur des architectures parallèles. C'est pourquoi nous en avons proposé une approximation, qui génère et évalue plusieurs hypothèses en parallèle. Ces deux opérations peuvent alors être traitées efficacement sur architectures parallèles. Les performances des différentes stratégies de filtrage particulières sont étudiées dans le chapitre 3. Néanmoins leurs particularités fondamentales méritent d'être notées dès maintenant :

- ▷ dans les méthodes de la première famille on lance en parallèle des hypothèses indépendantes, on les évalue en parallèle, et on propage les meilleures à l'observation suivante. Cette structure est adaptée à des calculs en parallèle, donc à des machines multi-processeurs.
- ▷ dans les méthodes de la seconde famille, le processus d'exploration de l'espace à un instant  $t$  est donc chaîné, ou itératif : il est plus « informé » que dans les filtres particuliers, puisque l'observation intervient dans la transition entre deux hypothèses.

Une deuxième typologie apparaît également, qui se révélera déterminante à l'examen des performances présentées dans le chapitre 3 :

- ▷ Le Filtre Particulaire *SIR* et le Filtre Particulaire *MCMC<sub>D</sub>* proposent leurs nouveaux échantillons en effectuant des mouvements dans toutes les dimensions de l'espace simultanément.
- ▷ A l'opposé le Filtre Particulaire *SIR Partitionné* et le Filtre Particulaire *MCMC<sub>d</sub>* proposent leurs nouveaux échantillons en effectuant des mouvements marginalisés selon un sous espace de dimension  $d$ .



---

# BANC D'ESSAIS POUR FILTRES PARTICULAIRES

---

Ce chapitre présente une étude expérimentale, comparant les performances de stratégies de ré-échantillonnage pour Filtres Particulaires fréquemment utilisées en suivi multi-objets. Nous présentons les résultats d'expériences et mettons en évidence les forces et faiblesses de chacune des approches. Nous montrons notamment que le Filtre Particulaire

par Chaîne de Markov (*FP MCMC*) est le plus performant, sous réserve que les propositions de mouvements de particules soient opérés dans un sous-espace de petite dimension. Nous montrons finalement que l'algorithme 2.8, le *FP MCMC* à propositions parallèles (présenté au chapitre 2) apporte des gains de performances considérables, par rapport au *FP MCMC* simple proposition.

---

### 3.1 Introduction

Nous avons argumenté en section 1.10 la nécessité de confier à un unique filtre le suivi joint de tous les objets à suivre. La conséquence est que le filtre opère dans un espace de grande dimension, que nous noterons  $D$ . L'approximation de la loi de probabilité *a posteriori* de l'état joint avec un nombre limité de particules est alors difficile, car cet état est décrit dans un espace de grande dimension, proportionnelle au nombre d'objets suivis. Le bon choix des échantillons à chaque instant, c'est à dire le bon rééchantillonnage des particules, est la clé de voûte d'un Filtrage Particulaire dans un tel espace de grande dimension. Parmi les plus utilisées dans les travaux publiés, nous avons choisi quatre stratégies d'échantillonnage, qui ont été présentées dans le chapitre 2 : le Filtre Particulaire SIR, le Filtre Particulaire SIR Partitionné, le Filtre Particulaire MCMC à mouvements joints, et le Filtre Particulaire MCMC à mouvements marginalisés. Dans ce chapitre, nous présentons un banc d'essais pour Filtres Particulaires, et un plan d'expériences visant à comparer les performances des quatre filtres présentés dans le chapitre 2. Ce plan d'expériences est un développement d'expérimentations publiées dans (BC08b). On verra qu'il est fructueux de marginaliser les propositions sur des sous-espaces de dimension notée  $d$ , inférieure à  $D$ . Sur les graphes de résultats, ces quatre filtres seront désignés comme suit :

- ▷ *FP SIR* pour le Filtre Particulaire SIR (Sequential Importance Resampling)
- ▷ *FP partition* pour le Filtre Particulaire partitionné
- ▷ *FP MCMC<sub>d</sub><sup>1</sup>* pour le Filtre Particulaire MCMC, l'indice  $d$  indiquant la dimension du sous-espace marginalisant les propositions, c'est à dire le nombre de composantes simultanément modifiées à chaque itération. Cet indice peut prendre toute valeur entière comprise entre 1 (on ne bouge alors qu'une composante à la fois) et  $D$  (on bouge alors toutes les composantes simultanément). L'exposant 1 désigne le Filtre Particulaire MCMC classique, où une seule proposition est générée et évaluée à chaque itération.
- ▷ *FP MCMC<sub>d</sub><sup>p</sup>* pour le Filtre Particulaire MCMC, l'indice  $d$  indiquant le nombre de dimensions simultanément modifiées à chaque itération. L'exposant  $p$  désigne l'extension parallélisée du Filtre Particulaire MCMC, où  $p$  propositions sont simultanément générées et évaluées par  $p$  processeurs à chaque itération.

On va voir apparaître dans ce chapitre des résultats très contrastés, qui découlent des deux typologies de filtres mentionnées dans la conclusion du chapitre 2. Ce plan d'expérience a été conçu de manière à déterminer lequel de ces quatre filtres est le plus adapté aux besoins de notre application de suivi Multi-Objets. Les conclusions que nous en tirons sont néanmoins généralisables aux besoins de tout filtrage temporel par méthode de Monte-Carlo, dans un espace d'état de grande dimension.

### 3.2 Méthode d'évaluation des performances des filtres

#### 3.2.1 Filtres Particulaires

Un Filtre Particulaire constitue un bon estimateur récursif de l'état d'un système, lorsque la fonction d'observation et/ou de prédiction sont quelconques (en particulier non nécessairement gaussiennes), cas général dans le suivi multi-objet. Nous avons cependant choisi de mener notre étude dans le cas particulier où la fonction d'observation et la fonction d'évolution sont gaussiennes, afin de permettre l'utilisation d'un Filtre de Kalman comme référence, car il est le filtre optimal dans ces conditions. Le besoin de cette référence est la seule raison du choix gaussien. Les filtres particuliers testés, eux, sont indifférents à cette hypothèse. Les performances obtenues sont donc généralisables en dehors de cette hypothèse. L'erreur commise par chacun des filtres est alors quantifiée par la Divergence de Kullback-Leibler entre les deux lois *a posteriori* respectivement estimées par le Filtre Particulaire et par le Filtre de Kalman.

#### 3.2.2 Filtre de Kalman

Un Filtre de Kalman délivre récursivement une forme paramétrique de la loi de probabilité *a posteriori* de l'état du système à l'instant  $t$ , à partir de la loi de probabilité *a posteriori* de l'état du système à l'instant  $t - 1$ , et de

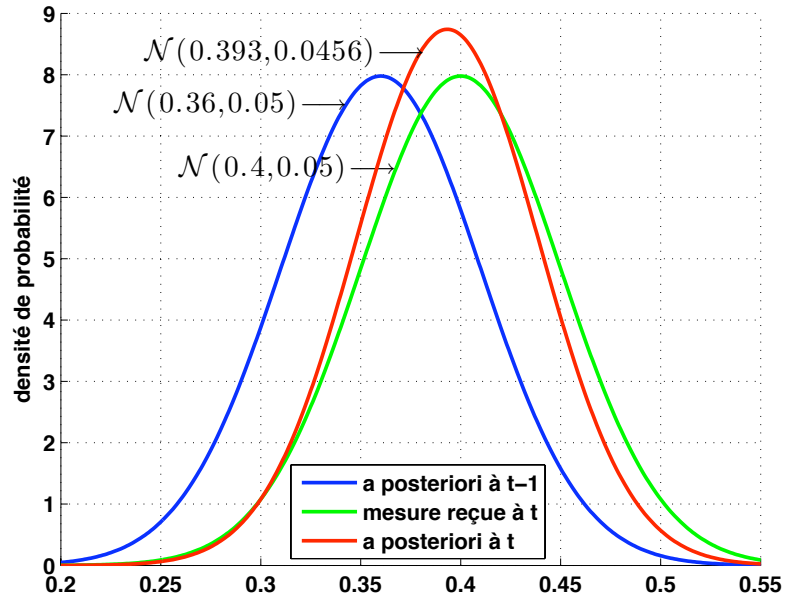


FIGURE 3.1 – Loi de probabilité *a posteriori* à l’instant  $t$  délivrée par Filtre de Kalman en dimension 1, à partir d’une loi de probabilité *a posteriori* à l’instant  $t - 1$  :  $\mathcal{N}(0.36, \sigma_{t-1}^{post} = 0.05)$ , d’une loi de probabilité de la mesure reçue à l’instant  $t$  :  $\mathcal{N}(0.4, \sigma_t^{mes} = 0.05)$ , et d’une loi de probabilité de dynamique de processus :  $\mathcal{N}(0, \sigma^{dyna} = 0.1)$ .

la loi de probabilité de la mesure reçue à l’instant  $t$ . La figure 3.1 illustre son comportement en dimension 1, dans les conditions suivantes :

- ▷ Une loi de probabilité *a posteriori* de l’état du système à l’instant  $t - 1$  normale de moyenne arbitrairement choisie à 0.36 et d’écart type  $\sigma_{t-1}^{post} = 0.05$ ,
- ▷ Une loi de probabilité de dynamique du système fixée :  $\mathcal{N}(0, \sigma^{dyna} = 0.1)$ ,
- ▷ Une loi de probabilité de mesure à l’instant  $t$  normale de moyenne arbitrairement choisie à 0.4 et d’écart type  $\sigma_t^{mes} = 0.05$ .

La figure 3.2 illustre les sensibilités respectives de la moyenne et de l’écart-type de la loi de probabilité de l’état du système *a posteriori*, délivrée par le Filtre de Kalman à l’instant  $t$ , aux variations de :

- ▷  $\sigma_{t-1}^{post}$  variant de 0 à 0.1,
- ▷  $\sigma_t^{mes}$  variant de 0 à 0.1.

Le point choisi pour tracer la figure 3.1 est tracé en noir sur les courbes. La densité de probabilité *a posteriori*, délivrée par le Filtre de Kalman à l’instant  $t$ , sera considérée comme la cible à atteindre par les autres filtres.

### 3.2.3 Filtre Particulaire contre Filtre de Kalman

La figure 3.3 compare le comportement d’un Filtre Particulaire avec celui d’un Filtre de Kalman. Bien entendu, pour pouvoir utiliser ce dernier, les lois doivent être gaussiennes. Elle illustre que dans ces conditions, la loi *a posteriori* à  $t$  estimée par un Filtre Particulaire présente les mêmes sensibilités à  $\sigma_{t-1}^{post}$  et à  $\sigma_t^{mes}$  que le Filtre de Kalman. Bien entendu, les résultats sont fort bruités avec un petit nombre de particules (en haut : 100 particules), du fait de l’aspect stochastique du Filtre Particulaire. L’augmentation du nombre de particules (milieu : 1000



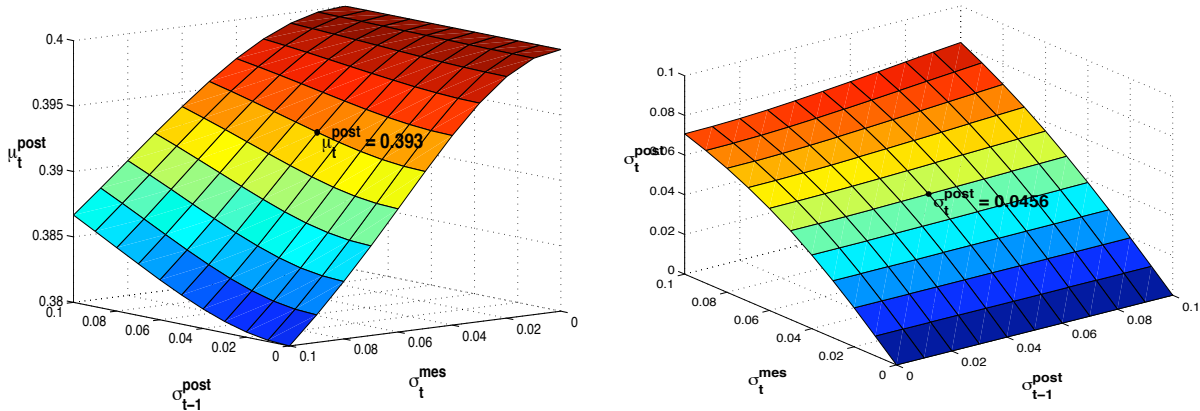


FIGURE 3.2 – Moyennes (à gauche) et écarts-types (à droite) de la loi *a posteriori* à  $t$  estimée à l'instant  $t$  par Filtre de Kalman, en dimension 1. Loi *a posteriori* à  $t - 1$  :  $\mathcal{N}(0.36, \sigma_{t-1}^{post} \in [0; 0.1])$ , densité de la mesure reçue à l'instant  $t$  :  $\mathcal{N}(0.4, \sigma_t^{mes} \in [0; 0.1])$ , densité dynamique de processus  $\mathcal{N}(0, \sigma = 0.1)$ .

particules, et bas : 10000 particules) lisse cet effet, permettant à la grille rouge (Filtre Particulaire) de converger vers la grille verte (Filtre de Kalman). Dans ces conditions, le Filtre Particulaire réalise une approximation échantillonnée d'un Filtre de Kalman.

### 3.3 Protocole d'expérimentation

Les quatre filtres testés seront comparés dans des espaces de dimensions variables, le Filtre de Kalman étant utilisé comme référence. L'étendue est bornée entre 0 et 1 dans toutes les dimensions. Les conditions d'expériences de notre campagne de tests résultent d'un choix arbitraire, mais représentatif d'un besoin en suivi Multi-Objets, c'est le point de référence défini dans la section 3.2.2 :

- ▷ On suppose que les mesure antérieures ont conduit tous nos filtres à estimer la loi *a posteriori* à  $t - 1$  par une loi normale  $\mathcal{N}(0.36, \sigma_{t-1}^{post} = 0.05)$ , de centre 0.36 et d'écart type 0.05, soit 5% de l'étendue de mesure dans chaque dimension.
- ▷ La loi de dynamique du système choisie est une loi normale  $\mathcal{N}(0, \sigma^{dyna} = 0.1)$  dans toutes les dimensions, soit 10% de l'étendue de mesure dans chaque dimension. C'est par cette loi que le filtre génère la loi *a priori* à  $t$ , dans la phase de prédiction. Cette loi est définie en fonction de la connaissance que l'on a de la dynamique du système.
- ▷ La fonction de vraisemblance des observations à l'instant  $t$  est arbitrairement choisie comme loi normale  $\mathcal{N}(0.4, \sigma_t^{mes} = 0.05)$  soit 5% de l'étendue de mesure dans chaque dimension.

#### 3.3.1 Nombre de particules, nombre d'évaluations

Si le Filtre de Kalman utilise directement la forme paramétrique des distributions décrites ci-dessus, les filtres testés ici s'appuient sur des échantillons pour représenter les distributions. Le nombre d'échantillons est le paramètre déterminant la performance d'un filtre de Monte-Carlo, et on verra que la sortie de tous les filtres converge vers celle du Kalman, mais ceci peut demander un nombre insupportable de particules, notamment dans les espaces de grande dimension.

En vision, l'opération la plus coûteuse est l'évaluation de la vraisemblance des particules, car elle est calculée sur l'image ou sur un descripteur de grande dimension. Le paramètre influant au premier ordre est le nombre

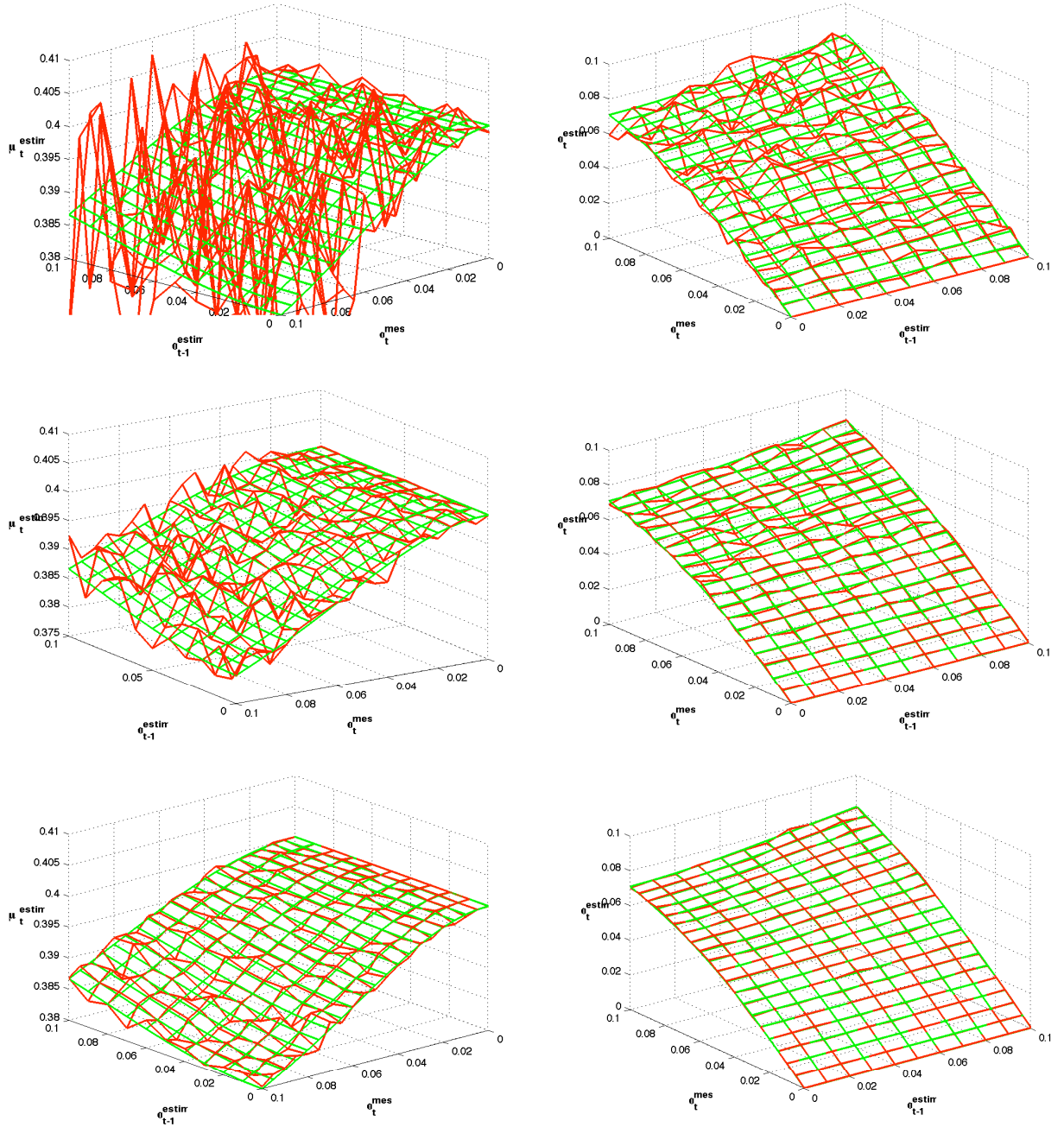


FIGURE 3.3 – Moyennes (à gauche) et écarts-types (à droite) de la loi *a posteriori* à  $t$  estimée à l'instant  $t$  par Filtre de Kalman (grille verte) et par Filtre Particulaire (grille rouge), en dimension 1, avec 100 particules (en haut), 1000 particules (milieu), et 10000 particules (bas). Loi *a posteriori* à  $t - 1$  :  $\mathcal{N}(0.36, \sigma_{t-1}^{post} \in [0; 0.1])$ , loi de la mesure reçue à l'instant  $t$  :  $\mathcal{N}(0.4, \sigma_t^{mes} \in [0; 0.1])$ , loi dynamique de processus  $\mathcal{N}(0, \sigma = 0.1)$ .

d'évaluations de vraisemblances de particules, requis par un rééchantillonnage. Une première campagne d'essais consistera alors à comparer les méthodes de rééchantillonnage à nombre d'évaluations de la fonction de vraisemblance constant. Ce nombre correspond au nombre de particules sur tous les filtres testés, sauf sur le *FP Partition*, qui évalue séquentiellement la vraisemblance d'une même particule dans chaque dimension. Un rééchantillonnage complet de ce filtre avec  $N$  particules dans un espace de dimension  $D$  requiert  $N * D$  évaluations. Nous avons cependant voulu étendre nos travaux au cas où le temps de calcul est déterminé par le

nombre de particules, et non par le nombre d'évaluations. Une deuxième campagne d'essais consistera alors à comparer les méthodes de rééchantillonnage à nombre de particules constant.

### 3.3.2 Distance entre lois de probabilité

Partant de la densité de probabilité *a posteriori* à l'instant  $t - 1$ , on laisse nos filtres opérer sur un pas temporel, afin que chaque filtre génère la loi de probabilité *a posteriori* à l'instant  $t$ . Le Filtre de Kalman utilisé comme référence, estime cette densité sous forme paramétrique par une gaussienne multivariée, notée  $\mathcal{N}_{FK}$ , et dont  $\mu_{FK}$  et  $\Sigma_{FK}$  sont respectivement le vecteur moyenne et la matrice de covariance de la distribution de probabilité *a posteriori* estimée par le Filtre de Kalman. Les Filtres Particulaires, eux, approximent cette loi par un nuage de particules discètes. Quantifier une distance entre une loi continue et une approximation échantillonnée n'est pas évident. On peut :

- ▷ Estimer la densité continue représentée par le nuage de particules, en leur appliquant un noyau (méthode KDE : Kernel Density Estimation). Mais la nature et la largeur du noyau sont arbitrairement choisies, ce qui peut biaiser considérablement les résultats.
- ▷ Construire un histogramme du nuage de particules, définissant une approximation discrète de sa densité, évaluer la première au centre de chaque maille de l'histogramme, et calculer la distance de Bhattacharyya entre ces deux lois. Ceci est efficace en petite dimension, mais n'est plus significatif dès que la dimension augmente, car l'histogramme devient éparé. A moins d'augmenter le nombre de particules en raison exponentielle de la dimension. Or nous voulons au contraire évaluer le comportement de nos filtres avec peu de particules dans des espaces de grandes dimensions. De plus le pas du maillage de l'histogramme est arbitraire, et son choix peut lui aussi biaiser considérablement les résultats.
- ▷ Puisque le nuage de particules approximait une gaussienne à l'instant  $t - 1$ , on peut le représenter à l'instant  $t$  par sa moyenne et sa covariance, puis calculer la Divergence de Kullback-Leibler entre cette loi paramétrique et celle estimée par Filtre de Kalman. Cette approche est valide car il a été démontré que approximations échantillonnées délivrées par les échantillonneurs utilisés convergent tous vers la loi cible, ici la loi *a posteriori* qui est gaussienne.

Nous retenons donc cette dernière solution pour notre plan d'expériences.

### 3.3.3 Divergence de Kullback-Leibler

Entre deux lois de probabilités  $P$  et  $Q$  définies par des densités notées  $p$  et  $q$ , sur un espace défini par le vecteur  $\mathbf{x}$ , la divergence de Kullback-Leibler de  $Q$  par rapport à  $P$ , est définie par :

$$D_{KL}(P\|Q) = \int_{-\infty}^{\infty} p(\mathbf{x}) \log \frac{p(\mathbf{x})}{q(\mathbf{x})} d\mathbf{x} \quad (3.1)$$

Si nos lois sont gaussiennes multivariées dans un espace de dimension  $D$ , on les note  $\mathcal{N}_{FP}$  et  $\mathcal{N}_{FK}$ . Elles représentent respectivement la loi définie par le nuage de particules après rééchantillonnage par un Filtre Particulaire, et la densité estimée par Filtre de Kalman. La Divergence de Kullback-Leibler de  $\mathcal{N}_{FK}$  par rapport à  $\mathcal{N}_{FP}$ , peut alors être exprimée par la forme analytique suivante :

$$D_{KL}(\mathcal{N}_{FP}\|\mathcal{N}_{FK}) = \frac{1}{2} \left( \log \left( \frac{|\Sigma_{FK}|}{|\Sigma_{FP}|} \right) + \text{tr}(\Sigma_{FK}^{-1} \cdot \Sigma_{FP}) + \Delta_{\mu}^T \cdot \Sigma_{FK}^{-1} \cdot \Delta_{\mu} - D \right) \quad (3.2)$$

Où  $\Sigma_{FP}$  et  $\Sigma_{FK}$  sont respectivement la matrice de covariance du nuage de particules après rééchantillonnage par le FP, et la matrice de covariance calculée par Filtre de Kalman,  $\Delta_{\mu}$  est le vecteur différence des moyennes tel que :

$$\Delta_{\mu} = \mu_{FK} - \mu_{FP} \quad (3.3)$$

Où  $\mu_{FP}$  et  $\mu_{FK}$  sont respectivement le vecteur moyenne du nuage de particules après rééchantillonnage par le FP, et le vecteur moyenne calculée par Filtre de Kalman. La figure 3.4 illustre la sensibilité de la Divergence de Kullback-Leibler en dimension 1.

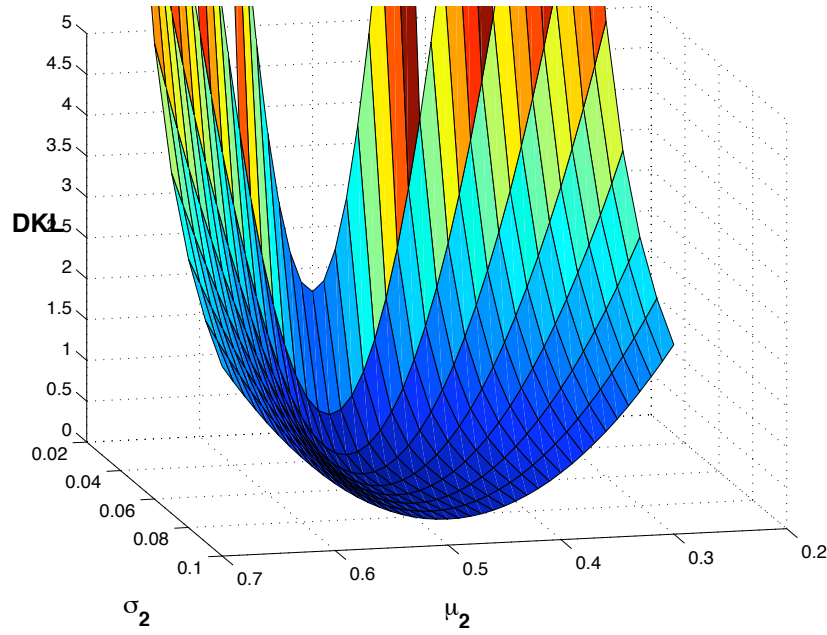


FIGURE 3.4 – Sensibilité de la Divergence de Kullback-Leibler entre deux lois gaussiennes en dimension 1. L'une des lois présente une moyenne et un écart-type arbitrairement fixés : respectivement  $\mu_1 = 0.5$  et  $\sigma_1 = 0.05$ . La sensibilité de la Divergence de Kullback-Leibler est présentée en faisant varier la moyenne et l'écart-type de la 2<sup>ème</sup> loi, respectivement notés  $\mu_2$  et  $\sigma_2$ .

### 3.3.4 Efficacité du rééchantillonnage

On veut optimiser l'efficacité du nuage de particules pour représenter au mieux les distributions, en évitant la dégénérescence des particules. La capacité du filtre à propager le plus grand nombre de particules distinctes, dans les régions intéressantes de l'espace d'état, doit donc être évaluée. Pour quantifier la qualité de l'échantillonnage d'une population de  $N$  particules, affectées de leurs poids  $\{\pi_n\}_{n=1}^N$  normalisés tels que  $\sum_{n=1}^N \pi_n = 1$ , un critère intéressant est proposé dans (ANN01) :

$$\eta = \left[ N \sum_{n=1}^N \pi_n^2 \right]^{-1} \quad (3.4)$$

Ce critère est très discriminant pour les méthodes parallèles : sa valeur est proche de 0 lorsque très peu de particules ont de gros poids, c'est à dire lorsque la région intéressante de l'espace d'état est sous-représentée. Sa valeur est proche de 1 lorsque toutes les particules ont des poids à peu près identiques, c'est à dire lorsqu'elles ont réussi à peupler la région intéressante de l'espace d'état. Malheureusement, ce critère n'est pas applicable directement aux méthodes itératives, où on propage seulement une population de  $N$  particules non pondérées. Rappelons que dans l'algorithme SIR, le nuage de particules pondérées  $\{\mathbf{X}_n, \pi_n\}_{n=1}^N$  avant rééchantillonnage est approximé par le nuage de particules pondérées  $\{\mathbf{X}_n^*, 1/N\}_{n=1}^N$  après rééchantillonnage. La probabilité de tirage de la particule  $\mathbf{X}_n$  est égale à son poids avant rééchantillonnage  $\pi_n$ . Après rééchantillonnage on se retrouve avec un nombre de copies  $\mathbf{X}_n^*$  de  $\mathbf{X}_n$  statistiquement proportionnel à ce poids  $\pi_n$ . Par contre le passage d'une représentation non pondérée à une représentation pondérée n'est pas évident. On a présenté en section 3.3.2 deux possibilités : KDE ou histogramme, mais aussi les inconvénients liés : il faut choisir arbitrairement soit la largeur du noyau, soit le pas de l'histogramme, ce qui peut biaiser considérablement les résultats. Pour comparer entre elles les méthodes itératives et les méthodes parallèles, nous adoptons comme critère d'efficacité de l'échantillonnage le taux d'unicité des l'échantillons  $\nu_u$  défini par :

$$\nu_u = \frac{\text{nombre de particules distinctes}}{N} \quad (3.5)$$

### 3.3.5 Critères de performances des Filtres Particulaires

La performance de nos Filtres Particulaires est alors évaluée selon trois critères :

- ▷ La Divergence de Kullback-Leibler de  $\mathcal{N}_{FK}$  par rapport à  $\mathcal{N}_{FP}$ , définie par l'équation (3.2).
- ▷ La déviation moyenne par dimension entre  $\mu_{FP}$ , moyenne du nuage de particules rééchantillonnées par le Filtre Particulaire, et  $\mu_{FK}$ , moyenne estimée par Filtre de Kalman :

$$\epsilon_P = \sqrt{\frac{1}{D} \sum_{i=1}^D \Delta_\mu(i)^2}, \quad (3.6)$$

où  $\Delta_\mu(i)$ ,  $i \in \{1, \dots, D\}$  désigne la  $i$ -ième composante de  $\Delta_\mu$ ,  $D$  étant la dimension de l'espace.

- ▷ Le taux d'unicité des l'échantillons  $\nu_u$ , défini par l'équation (3.5).

Les résultats obtenus seront discutés selon ces trois critères. Par contre, par souci de concision, seules les courbes de Divergence de Kullback-Leibler y figurent. Les courbes représentant la déviation moyenne par dimension et l'unicité des particules sont reportées dans l'annexe C.

## 3.4 Résultats

### 3.4.1 Effet du nombre d'évaluations

Dans ces essais, on travaille dans des espaces de dimensions arbitrairement fixées à 4 puis à 12, afin de mettre en évidence l'influence du nombre d'évaluations de la vraisemblance d'une particule, sur les critères de performance. L'expérience en espace de dimension 4 est menée pour caractériser le comportement des filtres dans un espace de petite dimension. La dimension 12, bien que modeste, suffira pour mettre en évidence les défaillances des filtres liées à l'accroissement de la dimension. Le nombre de particules utilisées est calculé de façon à nécessiter de 10 à 40000 évaluations, ce qui correspond au nombre de particules pour tous les filtres, sauf pour le Filtre Particulaire *FP Partition* : à nombre d'évaluations  $N$  constant, dans un espace de dimension  $D$ , il doit se contenter de  $N/D$  particules. Pour prendre en compte la nature stochastique des Filtres Particulaires, 20 répétitions des tests sont réalisées. La moyenne et l'écart-type sur ces répétitions sont présentés sur les courbes.

La figure 3.5 montre que tous les filtres à particules testés délivrent des distributions dont la moyenne converge bien vers celle estimée par le Filtre de Kalman, lorsque le nombre de particules est suffisant. Elle montre aussi que le Filtre Particulaire SIR est le plus performant (le plus proche des valeurs données par le Filtre de Kalman) dans un espace de petite dimension. La figure C.1 confirme ces constats sur la déviation des moyennes. La figure C.2 apporte un bémol à ce constat, car ce filtre délivre un taux d'unicité des particules bien faible : environ 20 %, contre plus de 40 % pour les Filtres Particulaires *FP Partition* et *FP MCMC*<sub>1</sub><sup>1</sup>. Notons que le *FP MCMC*<sub>D</sub><sup>1</sup> se comporte encore plus mal avec un taux d'unicité des particules de l'ordre de 5 % ! Ceci met déjà en évidence l'inefficacité d'un rééchantillonnage à mouvement joint dans toutes les dimensions.

Dans un espace d'état de dimension 12, les figures 3.6 et C.3 apportent un éclairage très différent : le Filtre Particulaire *FP MCMC*<sub>1</sub><sup>1</sup> se montre nettement plus performant que les autres. Les Filtres Particulaires *FP SIR* et *FP partition* offrent une performance bien inférieure au *FP MCMC*<sub>1</sub><sup>1</sup>, et l'avantage théorique du *FP Partition* est contrebalancé par la décroissance du nombre de particules lorsque la dimension de l'espace croît (car on est à nombre d'évaluations constant). Ils demandent beaucoup plus d'évaluations pour atteindre un même écart à la cible. Dans cet espace de dimension 12, le Filtre Particulaire *FP MCMC*<sub>D</sub><sup>1</sup> est complètement inopérant. Les nombres d'évaluations requis par tous les filtres sont évidemment plus grands en dimension 12 qu'en dimension 4.

Unicité des particules : la figure C.4 montre que les méthodes qui échantillonnent selon des mouvements marginaux (selon une seule dimension à la fois : *FP MCMC*<sub>1</sub><sup>1</sup> et *FP Partition*) maintiennent un taux d'unicité très supérieur aux méthodes qui échantillonnent selon des mouvements simultanés dans toutes les dimensions :



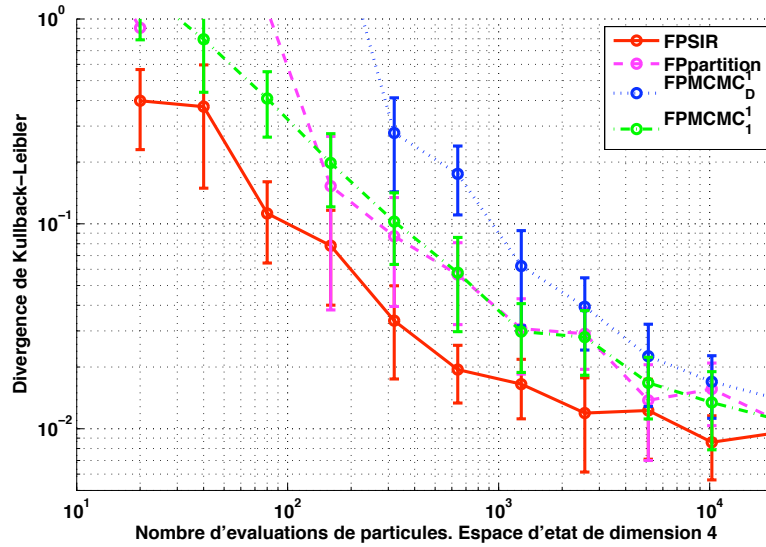


FIGURE 3.5 – Divergence de Kullback-Leibler entre la densité *a posteriori* délivrée par les Filtres Particulaires, et celle que délivre le Filtre de Kalman, en dimension 4. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

plus de 40 % pour *FP Partition* et *FP MCMC*<sub>1</sub><sup>1</sup>, contre 3 % pour le *FP SIR*. Le taux d'unicité très bas du *FP MCMC*<sub>D</sub><sup>1</sup> (moins de 1%), et tendant vers 0, indique que la chaîne est quasiment constituée de 1000 copies de la même particule !

### 3.4.2 Variation de la dimension de l'espace à nombre d'évaluations constant

Dans cette section, on fixe le nombre d'évaluations à 1000, et on fait varier la dimension de l'espace de 1 à 40, l'étendue de mesure étant toujours [0,1] dans toutes les dimensions. Comme dans la section précédente, ces résultats s'adressent aux applications où l'évaluation de la vraisemblance d'une particule est l'opération la plus coûteuse. Notre application en vision entre dans cette catégorie, car ce calcul fait intervenir l'image. Le nombre de particules est calculé de façon à nécessiter 1000 évaluations, ce qui correspond au nombre de particules pour tous les filtres, sauf pour le Filtre Particulaire *FP Partition* : à nombre d'évaluations  $N$  constant, dans un espace de dimension  $D$ , il doit se contenter de  $N/D$  particules. Pour prendre en compte la nature stochastique des Filtres Particulaires, 20 répétitions des tests sont réalisées. La moyenne et l'écart-type sur ces répétitions sont présentées sur les courbes.

Les résultats d'expériences présentés figures 3.7 et C.5 montrent des sensibilités à la dimension de l'espace très contrastées. Le Filtre Particulaire *FP MCMC*<sub>D</sub><sup>1</sup> commet une erreur d'estimation de la moyenne de 1% de l'étendue de mesure dès la dimension 4, le *FPSIR* : dès la dimension 7. Rappelons que la cible est la densité estimée par Filtre de Kalman : une moyenne *a posteriori* à l'instant  $t$  valant 0.393, alors que la moyenne de la densité *a posteriori* à l'instant  $t - 1$  était à 0.36. L'écart entre ces deux valeurs (0.033) est dépassé par ces deux filtres respectivement aux dimensions 8 et 13, ce qui signifie que dans ces dimensions, ces deux filtres deviennent incapables de rapprocher le nuage de particules de la densité cible ! La mauvaise répétabilité de ces deux filtres confirme qu'ils ne sont capables d'échantillonner correctement que dans des espaces de petites dimensions. A l'opposé le Filtre Particulaire *FP MCMC*<sub>1</sub><sup>1</sup> offre le meilleur comportement restant relativement proche du Kalman, et ce avec la meilleure répétabilité. Le Filtre Particulaire *FP Partition* est décevant, s'éloignant considérablement du Kalman dès la dimension 10. La raison est ici différente : comme on a fixé le nombre d'évaluations, son nombre de particules diminue quand la dimension augmente (il n'a plus que 25 particules en dimension 40 !). En résumé, le Filtre Particulaire *FP MCMC*<sub>1</sub><sup>1</sup> est celui qui « résiste le mieux à l'accroissement de la dimension de l'espace d'état ».

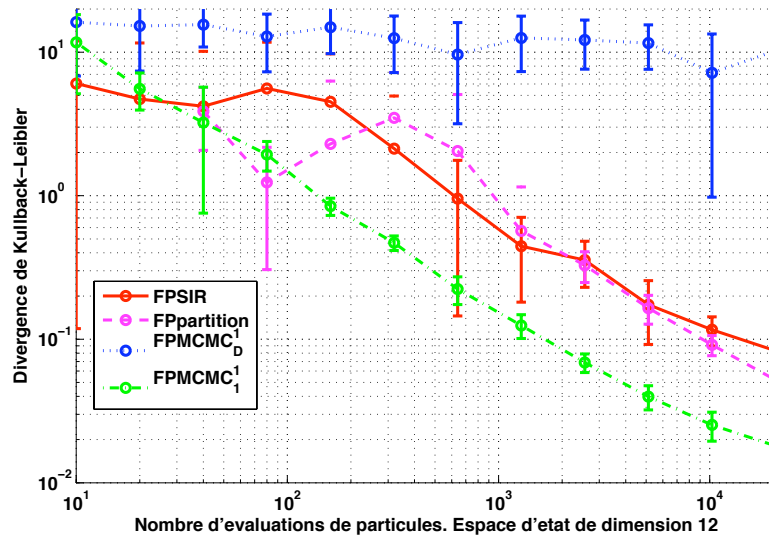


FIGURE 3.6 – Divergence de Kullback-Leibler entre la densité *a posteriori* délivrée par les Filtres Particulaires, et celle que délivre le Filtre de Kalman, en dimension 12. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

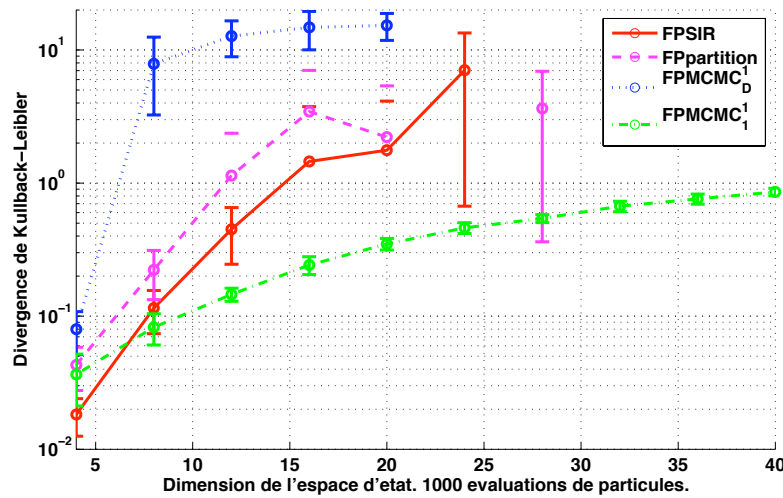


FIGURE 3.7 – Divergence de Kullback-Leibler entre la densité *a posteriori* délivrée par les Filtres Particulaires disposant de 1000 évaluations de leurs particules, et celle que délivre le Filtre de Kalman. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

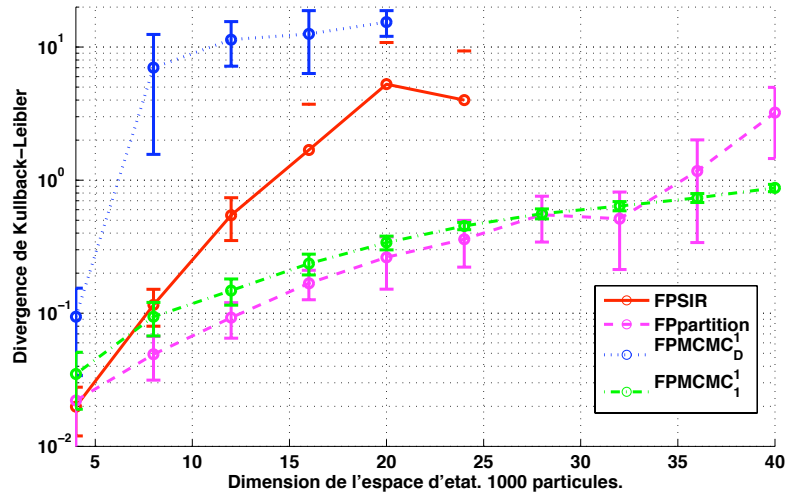


FIGURE 3.8 – Divergence de Kullback-Leibler entre la densité *a posteriori* délivrée par les Filtres Particulaires avec 1000 particules, et celle que délivre le Filtre de Kalman. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

### 3.4.3 Effet dimension à nombre de particules constant

Le classement des filtres présenté dans cette section est valable lorsque l'évaluation de la vraisemblance d'une particule n'est pas dominante sur les autres opérations. Dans ce cas, c'est le nombre de particules en présence qui détermine les coûts de calculs.

Les résultats d'expériences présentés figures 3.8 et C.7 montrent qu'avec un nombre de particules fixé, le  $FP\ MCMC_1^1$  et le  $FP\ Partition$  font jeu égal selon la déviation à la moyenne, et le classement est même favorable au  $FP\ Partition$  selon la divergence de Kullback-Leibler. Rappelons toutefois que le coût associé à cette méthode est de  $N * D$  évaluations, pour un filtre à  $N$  particules dans un espace de dimension  $D$ , et que cette méthode exige de disposer d'une fonction de vraisemblance marginale pour chaque dimension, condition parfois difficile à réaliser, notamment en vision. Le reste est inchangé.

Unicité des particules : la figure C.8 montre que les méthodes qui échantillonnent selon des mouvements marginaux (selon une seule dimension à la fois :  $FP\ MCMC_1^1$  et  $FP\ Partition$ ) maintiennent un taux d'unicité très supérieur aux méthodes qui échantillonnent selon des mouvements simultanés dans toutes les dimensions.

En conclusion, on retiendra que le Filtre Particulaire  $FP\ MCMC_1^1$  est le plus performant à nombre d'évaluations constant, dans un espace de dimension élevé.

### 3.4.4 Effet de la dimension du sous espace de propositions de particules

Il est clairement établi par les expériences présentées ci-dessus, qu'on accroît considérablement les performances d'un filtre, en explorant l'espace par succession de mouvements marginaux dans une seule dimension, plutôt que par des mouvements dans toutes les dimensions simultanément. L'expérience présentée dans cette section vise à définir le choix du sous-espace de mouvement maximisant la performance du filtre. Elle est focalisée sur le filtre Particulaire  $FP\ MCMC$ , dont on a vu les avantages ci-dessus. Il est testé dans trois version : la version présente dans les tests précédents, dénommée Filtre Particulaire  $FP\ MCMC_1^1$ , où les mouvements s'effectuent sur une seule dimension à la fois, une version dénommée Filtre Particulaire  $FP\ MCMC_4^1$ , où les mouvements s'effectuent sur 4 dimensions simultanément, enfin un Filtre Particulaire  $FP\ MCMC_D^1$ , où les mouvements s'effectuent sur toutes les dimensions simultanément. Les figures 3.9 et C.9 illustrent les sensibilités de ces filtres à la variation du nombre d'évaluations dans un espace de dimension arbitrairement fixée à 12. Elles confirment la très nette supériorité des deux versions marginalisant les mouvements : le Filtre Particulaire  $FP\ MCMC_1^1$  se classe un peu mieux que le Filtre Particulaire  $FP\ MCMC_4^1$ , tous deux étant très supérieurs au Filtre Particulaire  $FP\ MCMC_D^1$ . L'écart entre les deux premiers est faible. La figure C.10 établit néanmoins une



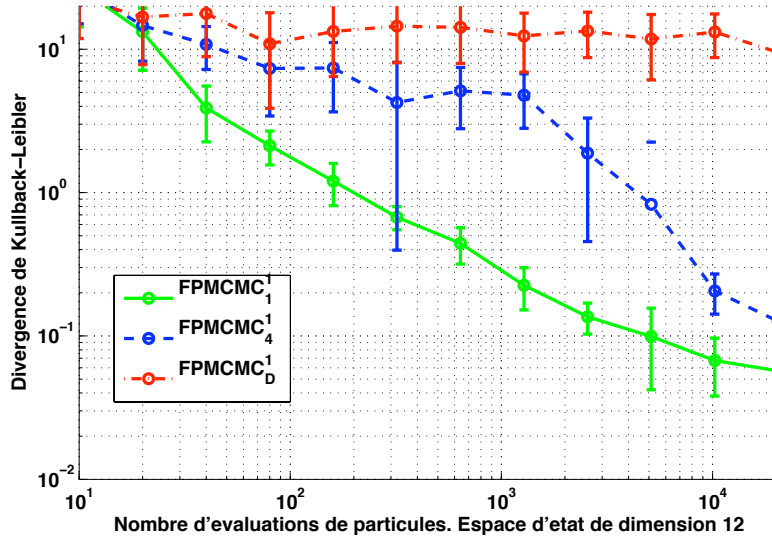


FIGURE 3.9 – Divergence de Kullback-Leibler entre la densité *a posteriori* approximée par les Filtres Particulaires, et celle que délivre le Filtre de Kalman, en dimension 12.

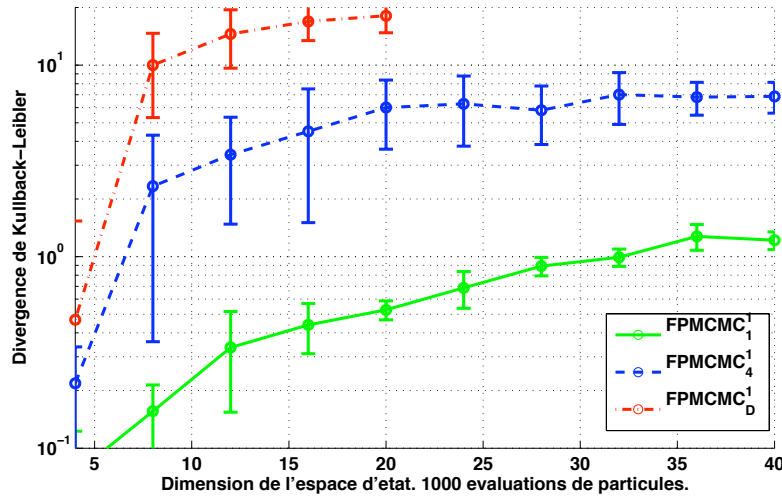


FIGURE 3.10 – Divergence de Kullback-Leibler entre la densité *a posteriori* délivrée par les Filtres Particulaires disposant de 1000 évaluations, et celle que délivre le Filtre de Kalman.

nette hiérarchie en faveur du Filtre Particulaire  $FP\ MCMC_1^1$ , qui assure un taux d'unicité des particules bien supérieur. Les figures 3.10, C.11, et C.12, illustrent les sensibilités de ces filtres à la variation de la dimension de l'espace, le nombre d'évaluations étant arbitrairement fixée à 1000. Ces graphes montrent que les écarts se creusent jusqu'à la dimension 10, puis se stabilisent. Les tendances observées plus haut sont ainsi confirmées quelle que soit la dimension de l'espace.

### 3.4.5 Composantes principales de la fonction de vraisemblance

La section précédente a mis en évidence l'intérêt de marginaliser à l'extrême les propositions, lorsque les composantes sont indépendantes sur toutes les dimensions. En effet, dans tous les tests menés jusqu'ici, la fonction de vraisemblance des observations à l'instant  $t$  était arbitrairement choisie comme loi normale  $\mathcal{N}(0.4, \sigma_t^{mes} = 0.05)$  dans chaque dimension. Cette loi était donc isotrope dans l'espace d'état, c'est à dire com-

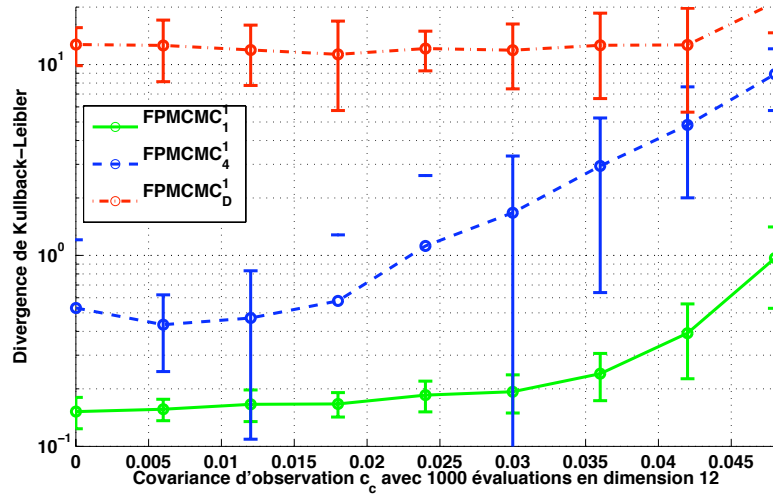


FIGURE 3.11 – Effet des covariances de couplage (hors diagonale) de la matrice de covariance de l’observation, notées  $c_c$ , sur la Divergence de Kullback-Leibler en dimension 12 entre la densité *a posteriori* délivrée par les Filtres Particulaires disposant de 1000 évaluations, et celle que délivre le Filtre de Kalman.

plètement découplée selon les dimensions. La matrice de covariance de cette fonction de vraisemblance des observations était alors diagonale dans la base de notre espace d’état, selon l’équation , avec  $c_d = (\sigma_t^{mes})^2 = 0.05^2$

$$\mathbf{C} = \begin{pmatrix} c_d & 0 & \dots \\ 0 & c_d & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} \quad (3.7)$$

Dans la réalité d’une application de suivi, toutes les composantes ne sont pas indépendantes, notamment les composantes d’un même objet entre elles. Dans le suivi d’un objet sur sol plan, la forme, la position, et la vitesse ne sont pas indépendamment observables. Pour évaluer l’effet de ces dépendances, nous expérimentons dans cette section l’effet d’une direction principale de l’observation, ne coïncidant pas avec les directions de la base de l’espace, sur les performances du filtre à propositions marginalisées. Pour ce faire, on fait varier figure 3.11 les covariances de couplage (hors diagonale) de la matrice de covariance de l’observation, notées  $c_c$  (3.8).

$$\mathbf{C} = \begin{pmatrix} c_d & c_c & \dots \\ c_c & c_d & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} \quad (3.8)$$

La figure 3.11 montre que le filtre à propositions jointes  $FP\ MCMC_D^1$  est peu sensible à cette direction principale, tandis que les performances des filtres à propositions marginalisées  $FP\ MCMC_4^1$  et  $FP\ MCMC_1^1$  se dégradent. Par conséquent, on aura intérêt à marginaliser les propositions dans des sous-espaces indépendants. Ce constat justifie le choix de propositions marginalisées dans le sous-espace décrivant un objet complet. Dans ce cas en effet, lorsque les objets n’interagissent pas entre eux, les sous-espaces de proposition deviennent indépendants. C’est d’ailleurs le choix qu’à notre connaissance, on rencontre dans la littérature.

### 3.4.6 Filtre Particulaire MCMC Parallélisé

Le  $FP\ MCMC$  à propositions parallèles a été présenté au chapitre 2, il est détaillé par l’algorithme 2.8. Il s’agit d’une extension du  $FP\ MCMC$  classique, où un seul mouvement est proposé à chaque itération de la chaîne. Les résultats suivants montrent que le  $FP\ MCMC$  à propositions parallèles permet des gains de performances considérables, par rapport au  $FP\ MCMC$  simple proposition, s’il est implanté sur machine à processeurs parallèles. Les figures 3.12 et C.13 montrent que l’on peut réduire considérablement le nombre de particules pour

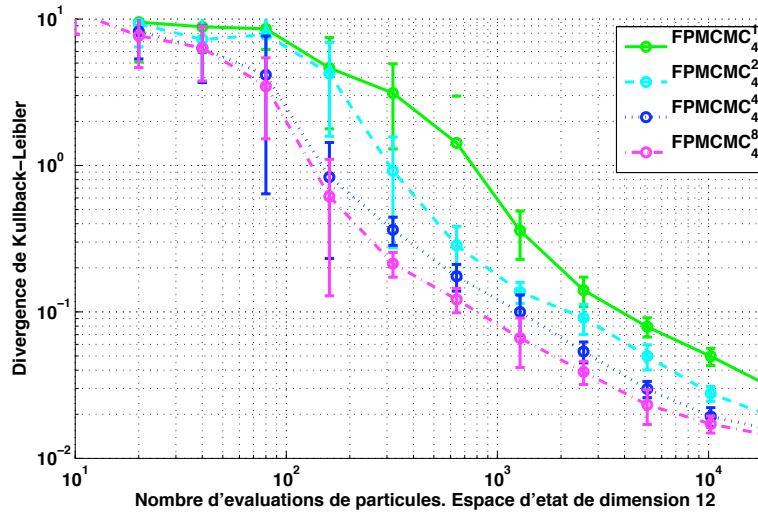


FIGURE 3.12 – Divergence de Kullback-Leibler entre la densité *a posteriori* délivrée par les Filtres Particulaires, et celle que délivre le Filtre de Kalman, en dimension 12.

une divergence donnée (de Kullback-Leibler ou de la moyenne). Les figures 3.13 et C.15 montrent que ces deux critères sont approximativement divisés par deux lorsqu'on lance en parallèle 8 propositions, tandis que le taux d'unicité est nettement rehaussé, comme l'atteste la figure C.16.

### 3.5 Conclusion

Dans ce chapitre, on a mené un plan d'expériences visant à comparer les performances de quatre Filtres Particulaires très utilisés, par rapport au Filtre de Kalman utilisé comme référence. Les expériences ont donc été menées dans un contexte où Kalman est opérationnel, c'est à dire en conditions gaussiennes.

- ▷ Dans les espaces de très petite dimension (de l'ordre de 4), le Filtre Particulaire SIR reste la solution la plus performante. Dès que la dimension augmente, ce filtre laisse dégénérer ses particules, c'est à dire qu'il ne propage qu'un petit nombre d'échantillons distincts, chacun recopié un grand nombre de fois, ce qui revient à gaspiller ces particules. Le Filtre Particulaire  $FP\ MCMC_D^1$  souffre du même défaut.
- ▷ Pour les espaces de dimension supérieure à 4, le Filtre Particulaire  $FP\ MCMC_1^1$  se montre de loin le plus performant : c'est lui qui estime la densité la plus proche de celle estimée par Filtre de Kalman, quelle que soit la dimension.
- ▷ Concernant le taux d'unicité des particules, deux filtres se montrent très supérieurs : le Filtre Particulaire  $FP\ Partition$  et le Filtre Particulaire  $FP\ MCMC_1^1$ , c'est à dire ceux qui proposent leurs nouveaux échantillons en effectuant des mouvements marginalisés selon une seule dimension.
- ▷ En absence de couplage entre composantes, les meilleures performances sont obtenues lorsque les propositions de mouvements de particules sont opérées dans un sous-espace de petite dimension. Idéalement de dimension  $d = 1$ , c'est à dire qu'on ne bouge qu'une composante à la fois.
- ▷ Lorsqu'on augmente les couplages entre composantes, les performances des filtres à propositions marginalisées se dégradent. Ce constat justifie le choix de propositions marginalisées dans le sous-espace décrivant un objet complet. Dans ce cas en effet, lorsque les objets n'interagissent pas entre eux, toutes les composantes couplées sont simultanément bougées.
- ▷ Les extensions du  $FP\ MCMC$  aux multiples propositions parallèles, montrent un gain de performance considérable, par rapport au  $FP\ MCMC$  standard, simple proposition.

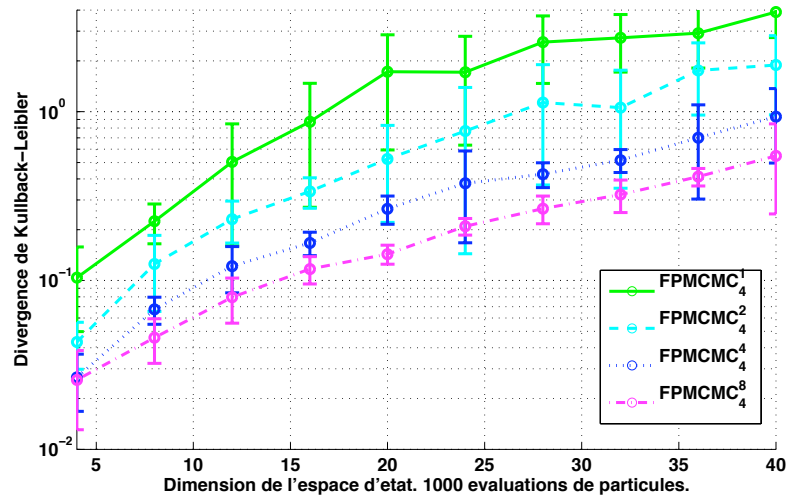


FIGURE 3.13 – Divergence de Kullback-Leibler entre la densité *a posteriori* délivrée par les Filtres Particulaires disposant de 1000 évaluations, et celle que délivre le Filtre de Kalman.

En résumé, le Filtre Particulaire  $FP\ MCMC_d^p$  constitue le meilleur choix pour répondre aux exigences du suivi multi-objets, avec  $p$  grand, et  $d$  égal à la dimension du descripteur d'un seul objet. Ce choix est en effet la plus petite valeur de  $d$  où toutes les composantes couplées sont simultanément bougées.



---

# SUIVI ET CATÉGORISATION D'OBJETS PAR FILTRE PARTICULAIRE *MCMC*

---

Ce chapitre décrit la mise en application du Filtre Particulaire *MCMC* au suivi et à la catégorisation en temps réel d'un nombre variable d'objets génériques. Différentes stratégies de filtrage particulaire ont été présentées au chapitre 2. A la lumière des résultats expérimentaux du chapitre 3, nous choisissons d'utiliser le  $FP\ MCMC_d^p$  pour le suivi multi-objets. L'indice  $d$  indique que les mouvements sont marginalisés sur un sous-espace de dimension  $d$ , et l'exposant  $p$  précise que l'on génère plusieurs propositions à chaque itération. L'algorithme est étendu pour suivre un nombre variable d'objets, on le nomme alors *FP RJ-MCMC* (Filtre Particulaire Reversible Jump Markov Chain Monte-Carlo). La méthode est suffisamment générique pour être adaptée au suivi de tout objet opaque sur sol plan. Elle a été utilisée sur les objets suivants : fourmis, piétons, motos, voitures, poids-lourds. Le modèle d'état, le modèle d'évolution, et le modèle d'observation alimentant ce filtre, sont décrits dans ce chapitre. Ces modèles ont été choisis pour répondre aux contraintes suivantes : nombre variable d'objets, fortes occultations inter-objets, fortes variation

de taille apparente des objets dues à la perspective, grandes disparité de dimensions réelles des objets suivis (moto contre semi-remorque dans le trafic routier), fortes variations d'éclairement. Nous avons délibérément choisi d'alimenter le filtre avec une observation pauvre, reposant uniquement sur une segmentation binaire *avant-plan / arrière-plan* basée sur un modèle de l'*arrière-plan* mis à jour en ligne à chaque image. Une fonction de vraisemblance basée sur cette segmentation *fond / forme*, relativement à la distance objet-caméra est proposée, ainsi qu'une modélisation des interactions entre objets. Afin d'améliorer la cohérence temporelle de présence et d'identité des objets, nous proposons de guider les sorties d'objets au moyen de la *vitalité*, une variable compilant l'historique de chaque objet. Pour suivre différentes catégories d'objets, nous proposons de confier au *Filtre Particulaire RJ-MCMC* les tâches conjointes de suivi et la catégorisation des objets. Pour améliorer la robustesse du suivi aux variations de conditions d'illumination, nous proposons de modéliser la source lumineuse et de confier son suivi dynamique au *Filtre Particulaire RJ-MCMC*, conjointement aux objets.

---

## 4.1 Introduction

Le suivi d'un nombre fixe d'objets par Filtre Particulaire *MCMC* a été proposé par Khan *et al.* ([KBD04](#)). La particularité du Filtre Particulaire *MCMC* est de confier le rééchantillonnage des particules à un échantillonneur par chaîne de Markov selon l'algorithme de Metropolis-Hastings ([Mac03](#)). Pour permettre aux objets d'entrer et sortir de la scène, Khan *et al.* ont étendu le *FP MCMC* au suivi d'un nombre variable d'objets préalablement détectés ([KBD05](#)). Pour ce faire, l'échantillonnage est mené par un échantillonneur *RJ-MCMC* (Reversible Jump Markov Chain Monte Carlo), proposé par Green ([Gre95](#)). Khan *et al.* ont proposé une implémentation du *FP RJ-MCMC* qui prend en compte comme observation d'entrée la sortie d'un détecteur d'objets (des fourmis) ([KBD05](#)). Cette approche convient bien sur la scène quasi-bidimensionnelle du suivi de fourmis, mais un détecteur d'apparence d'objets est vite mis en défaut en cas d'occultations. Dans notre implémentation, pour gérer ces occultations, nous avons choisi d'exploiter l'observation globale bas niveau de l'image : une carte binaire *fond / forme*. Il n'y a donc pas de détecteur en amont du filtre, et on peut considérer que l'étape de détection d'objet est intégrée au filtre. L'inconvénient de cette approche est que les occultations rendent cette observation globale impossible à partitionner en scores individuels de chaque objet, aussi l'exploration se fait sur l'espace d'état joint, avec une observation jointe.

La fonctionnalité principale d'un système de suivi d'un nombre variable d'objets est sa capacité à estimer une configuration globale multi-objets, c'est à dire le nombre d'objets présents dans le champ de mesure, et leurs positions respectives. La plus-value apportée par l'utilisation d'un filtre est la cohérence temporelle, non seulement des positions, mais aussi du nombre d'objets présents et de leurs identités respectives. La cohérence temporelle des positions est apportée par le choix de lois de prédiction pertinentes par rapport à la dynamique des objets suivis. La cohérence temporelle du nombre d'objets présents dans le champ de vision et de leurs identités respectives est par contre beaucoup plus délicate à modéliser, car elle dépend du champ de vision et des possibilités d'occultations des objets par le fond et par d'autres objets. En effet les mouvements d'entrée et de sortie d'un objet proposés par Khan *et al.* dans leur *FP RJ-MCMC* présentent l'inconvénient de permettre à un objet de sortir alors qu'il est dans le champ de mesure, et que son observabilité est provisoirement mauvaise, car mal segmenté du fond ou occulté par le fond ou par un autre objet. Lorsque cet objet redevient observable, le *FP RJ-MCMC* va proposer et accepter l'entrée d'un nouvel objet, muni d'une nouvelle identité. On a ainsi perdu l'identité initiale de l'objet. Dans le cas où chaque objet présente une apparence bien distincte des autres, il suffirait d'injecter dans l'observation un descripteur adapté pour discriminer les objets et associer la nouvelle identité à celle de l'objet sorti qui lui ressemble le plus. Malheureusement, dans nos applications de suivi de piétons et de véhicules, cette solution s'avère inefficace, car les apparences des objets sont souvent très proches. Ce problème est encore plus aigu dans l'application de suivi de fourmis adressé par Khan *et al.* . C'est pourquoi ils ont proposé l'adjonction d'une paire de mouvements supplémentaires : *stay* et *leave*, dans le but d'améliorer la cohérence temporelle du nombre d'objets présents dans le champ de vision et de leurs identités respectives ([KBD05](#)). Les expérimentations présentées au chapitre 5 montrent que cette paire de mouvement est efficace lorsque l'objet est peu observable pendant un faible nombre d'images, mais se montre insuffisante lorsque ce nombre augmente. Nous proposons dans ce chapitre d'affecter à chaque objet une variable additionnelle, la *vitalité*, mise à jour sur la base des observations. Elle permet de rendre réglable le nombre d'observations nulles auquel un objet survivra et conservera son identité.

En dehors de la fonction d'observation et de la *vitalité*, la méthode présentée dans ce chapitre est basée sur les travaux de Khan *et al.* , que nous étendons aux fonctionnalités suivantes :

- ▷ Suivi et catégorisation conjoints des objets : dans les applications où plusieurs catégories d'objets évoluent simultanément, et selon l'observation choisie, il peut s'avérer impossible de mener le suivi de l'une d'elles en ignorant les autres. C'est le cas en suivi de véhicules par segmentation *fond / forme*. Cette fonctionnalité est alors une condition nécessaire au simple suivi.
- ▷ Suivi conjoint des objets et du soleil pour modéliser les ombres portées des objets. Dans les applications en extérieur, les ombres portées perturbent considérablement la segmentation *fond / forme*, donc le suivi. Pour surmonter cette difficulté, nous avons choisi de modéliser l'éclairement et les ombres portées qui en résultent. Confier à l'utilisateur la modélisation de la source d'éclairement est contraignant, voire

impossible lorsque la météo est variable (alternance de nuages et d'éclaircies). Aussi l'option choisie consiste à confier au filtre le suivi conjoint de l'éclairement et des objets.

- ▷ Utilisation systématique du  $FP\ MCMC_d^p$  à propositions multiples (présenté au chapitre 2) pour accroître les performances. Pour épargner au lecteur la lecture d'algorithmes trop lourds, nous les présenterons dans la version simple proposition  $FP\ MCMC_d^1$ .

Notre implémentation suit jusqu'à une quinzaine d'objets au temps réel vidéo (25 images par seconde). Elle a été initialement présentée dans (BC08a), exposant l'intérêt d'associer le *FP RJ-MCMC* à une fonction de vraisemblance prenant en compte la distance des objets à la caméra. L'extension multi-propositions du *FP RJ-MCMC* a été proposée dans (BCR09b). Le suivi et la classification conjoints des objets, ont été publiés dans (BCR09c). Enfin la modélisation et le suivi conjoint de l'éclairement et des objets par le *FP RJ-MCMC*, toujours associé à la classification des objets, ont été publiés dans (BCR09a). Le système de suivi et catégorisation multi-objets présenté dans ce chapitre repose sur le filtrage des observations délivrées par une ou plusieurs caméras, par un Filtre Particulaire de type  $FP\ MCMC_d^p$ . Cette technique a été présentée au chapitre 2, et a été validée au chapitre 3 sur des données de synthèse pour approximer récursivement à chaque instant la loi de probabilité *a posteriori* dynamique d'un état décrit dans un espace de grande dimension. Il ne s'agissait que de données de synthèse, portant notamment sur des variables continues, indépendantes entre elles. Le passage au suivi réel d'objets matériels pouvant entrer ou sortir du champ de mesure, et pouvant interagir entre eux, demande les extensions suivantes :

- ▷ La dimension de l'espace d'état décrivant la configuration globale multi-objets doit pouvoir s'adapter au nombre d'objets présents dans le champ de mesure.
- ▷ Pour obtenir des performances satisfaisantes tant en suivi qu'en temps de calcul, il est nécessaire de modéliser les interactions entre objets, notamment l'impossibilité matérielle que deux objets rigides s'interpénètrent. Les composantes du vecteur d'état ne sont plus toutes indépendantes les unes des autres.
- ▷ Certaines variables du vecteur d'état seront discrètes : la catégorie de chaque objet, et l'état du soleil (masqué ou non par un nuage).

Ce chapitre est organisé ainsi : en section 4.2, nous appliquons l'algorithme 2.7 présenté au chapitre 2, au suivi d'un nombre fixe d'objets. Les extensions nécessaires pour suivre un nombre variable d'objets sont présentées en section 4.3. Notre modèle d'observation est présenté en section 4.4. Cette section présente une mesure de vraisemblance basée sur la segmentation *fond / forme* originale, relative à la distance objet-caméra. Le guidage des mouvements d'entrée et de sortie à partir de l'observation et de la *vitalité* de chaque objet, est présenté respectivement dans les sections 4.5 et 4.6. La section 4.7 synthétise la prise en compte de ces mouvements. Dans la section 4.8, on présente la modélisation des interactions entre objets, et leur prise en compte dans le filtrage. La modélisation est basée sur une distance anisotrope entre leurs modèles géométriques 3D. La section 4.9 étend le modèle d'état, afin de suivre plus efficacement des objets dont la dynamique est connue, tels que les véhicules terrestres. Nous étendons en section 4.10 la méthode *FP RJ-MCMC*, pour permettre au filtre de mener conjointement le suivi et la catégorisation d'un nombre variable d'objets. Enfin, en section 4.11, nous présentons la modélisation des ombres et le suivi de l'illumination.

## 4.2 Suivi d'un nombre fixe d'objets par *FP MCMC*

### 4.2.1 Modèle d'état de la configuration jointe multi-objets

Nous avons argumenté au chapitre 1 la nécessité du suivi par filtrage joint de la configuration globale multi-objets, c'est à dire qu'un unique filtre estime récursivement la loi de probabilité de l'état global des objets. Le suivi d'objets consiste à estimer cette configuration jointe multi-objets, ou état, noté  $\mathbf{X}_t$  à chaque instant  $t$ . Les résultats expérimentaux présentés en annexe A illustrent et quantifient le gain en précision du suivi apporté par



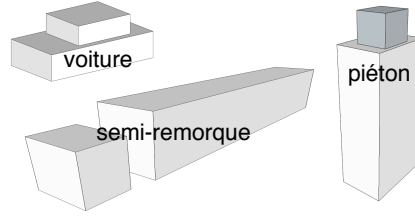


FIGURE 4.1 – Modèles géométriques d'une voiture, d'un camion semi-remorque et d'un piéton par 2 parallélépipèdes.

un suivi par filtre joint, comparé à plusieurs filtres disjoints. Afin de suivre simultanément un nombre variable d'objets, l'état encode le nombre et la configuration des objets perceptibles dans le champ de mesure :

$$\mathbf{X}_t^n = \{J_t^n, \mathbf{x}_t^{j,n}\}, \forall j \in \{1, \dots, J_t^n\}, \quad (4.1)$$

où  $J_t^n$  est le nombre d'objets visibles contenus dans la particule  $n$  à l'instant  $t$ , et chaque objet  $j$  est défini par :

$$\mathbf{x}_t^{j,n} = \{\mathbf{p}_t^{j,n}, \mathbf{v}_t^{j,n}, \mathbf{s}_t^{j,n}\} \quad (4.2)$$

La position absolue de l'objet  $j$  de la particule  $n$  au temps  $t$ , est définie par  $\mathbf{p}_t^{j,n} = (x_t^{j,n}, y_t^{j,n}, \psi_t^{j,n})$ , où  $x_t^{j,n}$  et  $y_t^{j,n}$  est la position absolue du centre de gravité et  $\psi_t^{j,n}$  est l'angle de cap. La vitesse absolue de l'objet est  $\mathbf{v}_t^{j,n}$ , un vecteur décrit par son intensité et son orientation. La forme de l'objet peut être modélisée par un nombre choisi de parallélépipèdes définis par des dimensions et positions relatives. La figure 4.1 représente la modélisation géométrique d'une voiture, d'un camion semi-remorque et d'un piéton par l'union de deux parallélépipèdes. Nous choisissons une modélisation plus simple, par un seul parallélépipède. Ainsi  $\mathbf{s}_t^{j,n}$  regroupe la longueur, la largeur et la hauteur de l'objet.

### 4.2.2 Modèle d'évolution dynamique des objets

Afin d'améliorer l'efficacité du suivi, nous limitons l'espace d'état à l'étendue que peuvent physiquement atteindre les objets suivis. A cette fin, chaque objet est contrôlé par le modèle dynamique correspondant à sa catégorie. Si l'objet est supposé être un piéton, sa dynamique est décrite par les équations (4.3) et (4.4) :

$$x_t^{j,n} = x_{t-1}^{j,n} + T \cdot v_{t-1}^n \cdot \cos(\psi_{t-1}^{j,n}), \quad (4.3)$$

$$y_t^{j,n} = y_{t-1}^{j,n} + T \cdot v_{t-1}^n \cdot \sin(\psi_{t-1}^{j,n}), \quad (4.4)$$

où  $v_{t-1,i}^n$  est la norme de la vitesse de l'objet au temps  $t - 1$ , et  $T$  est la période d'acquisition vidéo.

### 4.2.3 Mise en œuvre du *FP MCMC*

L'algorithme 4.1 détaille le processus de suivi d'un nombre fixe d'objets. Il est simplement l'adaptation de L'algorithme 2.7 (chapitre 2), dans lequel chaque sous espace marginalisé représente un objet. Les résultats expérimentaux du chapitre 3 montrent que cette marginalisation des mouvements est cruciale pour les performances. En suivi multi-objets, elle se fait sur le sous-espace représentant un objet, comme l'ont suggéré les résultats du chapitre 3. A chaque itération, on change toutes les dimensions d'un seul objet à la fois, tiré aléatoirement parmi  $\{1, \dots, J_t^{n-1}\}$ ,  $J_t^{n-1}$  étant le nombre d'objets de la particule  $n - 1$ . Les autres objets restent inchangés. La figure 4.2 illustre le processus.

On a vu au chapitre 2 que le calcul du taux d'acceptation requiert à chaque itération les lourds calculs décrits par l'équation 2.23, c'est à dire que la simplification qui nous avait conduit à l'équation 2.12 dans le cas du filtre à mouvements non marginalisés noté *FP MCMC<sub>D</sub>*, n'est plus possible le cas du filtre à mouvements marginalisés, noté *FP MCMC<sub>d</sub>*. Notre priorité étant que notre système de suivi tourne au temps réel vidéo, nous avons néanmoins décidé pour nos expérimentations, de faire une infidélité à l'algorithme du *FP MCMC*, en calculant le taux d'acceptation selon l'équation 2.12, au risque de biaiser le suivi.

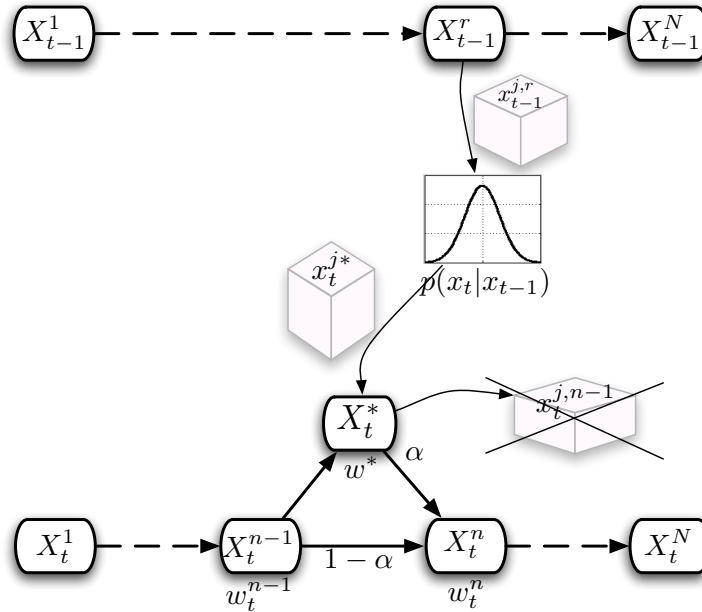


FIGURE 4.2 – Suivi d'un nombre fixe d'objets. Construction de la chaîne à l'instant  $t$ , à partir de la chaîne à l'instant  $t - 1$ . Une itération du filtre *FP MCMC* produit une proposition  $\mathbf{X}_t^*$ , en remplaçant l'objet  $\mathbf{x}_t^{j,n-1}$  de la particule  $\mathbf{X}_t^{n-1}$ , par l'objet  $\mathbf{x}^*$ . Celui-ci a été créé par tirage dans la loi dynamique  $p(\mathbf{x}_t|\mathbf{x}_{t-1})$  à partir de l'objet  $\mathbf{x}_{t-1}^{j,r}$ , extrait de la particule  $\mathbf{X}_{t-1}^r$ , tirée au hasard dans la chaîne à  $t - 1$ . On génère ainsi une nouvelle particule hybride  $\mathbf{X}_t^*$  qui sera acceptée comme prochaine itération  $\mathbf{X}_t^n$  avec la probabilité  $\alpha$ . Si elle est rejetée alors on duplique la précédente :  $\mathbf{X}_t^n = \mathbf{X}_t^{n-1}$ .

#### 4.2.4 Estimation de l'état de la configuration jointe multi-objets.

On a vu au chapitre 2 qu'à chaque pas temporel, les *FP* parallèles type *FP SIR* approximent la loi de probabilité *a posteriori*  $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$  de l'état  $\mathbf{X}_t$  à l'instant  $t$ , conditionné à l'historique des mesures  $\mathbf{Z}_{1:t}$ , par un nuage de  $N$  particules pondérées  $\{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N$  avec les poids associés  $\pi_t^n \propto P(\mathbf{Z}_t|\mathbf{X}_t^n)$ . Lorsqu'on le souhaite, on peut tirer de ce nuage une estimation de cet état. Cette estimation peut par exemple être calculée comme l'espérance de  $\mathbf{X}_t$ , approximée par  $\bar{\mathbf{X}}_t = \sum_{n=1}^N \pi_t^n \mathbf{X}_t^n$ , comme proposé dans (MA98). Ce choix pose problème lorsque la loi présente plusieurs modes, car il peut générer une estimation « fantôme », centrée sur le barycentre de ces modes, plutôt que sur l'un des modes. Pour éviter ce phénomène, une alternative est de prendre le *Maximum A Posteriori* (MAP), dont une approximation est donnée par la particule de meilleur poids. D'un autre côté, les Filtres Particulaires chaînés tels que le *FP MCMC* approximent  $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$  par un ensemble de  $N$  particules non pondérées  $\{\mathbf{X}_t^n\}_{n=1}^N$ . Dans ce cas aussi, l'estimation peut être calculée comme l'espérance de  $\mathbf{X}_t$ , approximée par  $\bar{\mathbf{X}}_t = \sum_{n=1}^N \mathbf{X}_t^n$ , avec le même problème que précédemment lorsque la loi présente plusieurs modes. On a vu en section 2.2 que l'approximation pondérée  $\{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N$  et l'approximation non pondérée  $\{\mathbf{X}_t^n\}_{n=1}^N$  sont deux approximations duales de la même loi de probabilité, où les nombres de copies de chaque particule dans la représentation non pondérée correspondent aux poids des particules dans la représentation pondérée. Dans le cas non pondéré, une approximation du *Maximum A Posteriori* consiste donc à choisir la particule la plus recopiée. C'est la solution que nous adoptons.

### 4.3 Suivi d'un Nombre Variable d'Objets par *FP RJ-MCMC*

Pour permettre aux objets d'entrer et sortir de la scène, Khan *et al.* ont étendu le *FP MCMC* au suivi d'un nombre variable d'objets (KBD05). Pour ce faire, le rééchantillonnage est mené par l'algorithme *FP RJ-MCMC* (Reversible Jump Markov Chain Monte-Carlo), proposé par Green (Gre95). Comme le nombre d'objets visibles peut changer, la dimension de l'espace d'état peut aussi changer, l'état peut « sauter » d'un sous espace d'état

**Algorithme 4.1** Suivi d'un Nombre Fixe d'Objets par *FP MCMC***Entrées :**

- nuage de particules approximant la loi de probabilité *a posteriori* de l'état joint multi-objets  $\mathbf{X}_{t-1}$  à l'instant  $t - 1$  :  $\{\mathbf{X}_{t-1}^n\}_{n=1}^N$ , tel que  $\mathbf{X}_{t-1}^n = [\mathbf{x}_{t-1}^{j,n}]_{j=1}^{J_{t-1}^n}$ , où  $\mathbf{x}_{t-1}^{j,n}$  est le descripteur de l'objet  $j$ , de dimension  $d$ ,  $\forall j \in \{1, \dots, J_{t-1}^n\}$ .  $J_{t-1}^n$  est le nombre d'objets décrits par la particule  $n$ .
- lois de probabilité de prédiction marginales  $p(\mathbf{x}_t^j | \mathbf{x}_{t-1}^j)$ , modélisant la dynamique de l'objet  $j$ .
- modèle de vraisemblance globale d'observation  $p(\mathbf{Z} | \mathbf{X})$ .
- observation  $\mathbf{Z}_t$  à l'instant  $t$ .

**Initialisation de la chaîne :**

- Tirer aléatoirement une particule de la chaîne précédente :  $\mathbf{X}_{t-1}^r$ ,  $r \in \{1, \dots, N\}$ .
- Générer une particule initiale  $\mathbf{X}_t^0 = [\mathbf{x}_t^{j,0}]_{j=1}^{J_{t-1}^r}$ , où les  $\mathbf{x}_t^{j,0} \forall j \in \{1, \dots, J_{t-1}^r\}$  sont tirées des lois de prédiction marginales  $p(\mathbf{x}_t^j | \mathbf{x}_{t-1}^j)$ .
- Évaluer son poids  $w_t^0 \propto P(\mathbf{Z}_t | \mathbf{X}_t^0)$ , vraisemblance de l'observation  $\mathbf{Z}_t$  sachant l'état  $\mathbf{X}_t^0$ .

**pour**  $n = 1$  à  $N + N_B$  **faire**

- Tirer l'indice de l'objet à bouger :  $j \in \{1, \dots, J_{t-1}^{n-1}\}$
- Propagation : tirer aléatoirement une particule de la chaîne précédente :  $\mathbf{X}_{t-1}^r$ ,  $r \in \{1, \dots, N\}$ , et extraire l'instance d'objet  $j$  qu'elle contient :  $\mathbf{x}_{t-1}^{j,r}$
- Prédiction : générer  $\mathbf{x}_t^{j*} \sim p(\mathbf{x}_t^j | \mathbf{x}_{t-1}^{j,r})$ , proposition d'objet  $j$  tirée de la loi marginale dynamique.
- Générer une particule prédite  $\mathbf{X}_t^*$ , en remplaçant l'instance d'objet  $j$  notée  $\mathbf{x}_t^{j,n-1}$  de  $\mathbf{X}_t^{n-1}$ , par  $\mathbf{x}_t^{j*}$ .
- Évaluation : calculer  $w^* \propto P(\mathbf{Z}_t | \mathbf{X}_t^*)$ , vraisemblance de l'observation  $\mathbf{Z}_t$  sachant l'état joint  $\mathbf{X}_t^*$ .
- Calcul du taux d'acceptation selon l'équation (2.23).
- Ajout du  $n^{\text{ième}}$  élément de la chaîne :  $\mathbf{X}_t^n = \mathbf{X}_t^*$  et  $w_t^n = w^*$  avec la probabilité  $\alpha$ , ou duplication du précédent  $\mathbf{X}_t^n = \mathbf{X}_t^{n-1}$  et  $w_t^n = w_t^{n-1}$  avec la probabilité  $1 - \alpha$ .

**fin pour**

**Burn-in** : supprimer les  $N_B$  premiers échantillons de la chaîne.

**Sortie** : nuage de particules approximant la loi de probabilité *a posteriori* de  $\mathbf{X}_t$  à l'instant  $t$  :

$$\{\mathbf{X}_t^n\}_{n=N_B+1, \dots, N_B+N}$$

vers un autre sous espace d'état de plus grande ou plus petite dimension si un nouvel objet entre ou si un objet quitte la scène. L'exploration de cet espace doit donc elle aussi pouvoir « sauter » d'un sous-espace d'état vers un autre sous-espace. Pour empêcher la chaîne de recherche de se bloquer dans un minimum local, les sauts entre les sous-espace doivent être réversibles. Pour cette raison, Khan *et al.* ont proposé la paire de mouvements discrets réversibles *{entrée, sortie}* (KBD05). La fonction du filtre multi-objets est alors d'estimer récursivement la configuration globale, incluant les identifiants des objets présents dans le champ de perception. On confie ainsi au *FP RJ-MCMC* l'échantillonnage d'une loi *a posteriori* qui est définie sur l'union de plusieurs sous-espaces de dimensions différentes. Si on veut que l'échantillonnage converge toujours vers la loi *a posteriori*, il faut de plus que le calcul du taux d'acceptation reste fidèle au principe de l'algorithme de Metropolis-Hastings.

### 4.3.1 Mouvements proposés

Dans sa version de base, le *FP RJ-MCMC* met en œuvre trois types de mouvements affectant la configuration jointe multi-objets : l'entrée d'un nouvel objet absent dans la particule précédente, la mise à jour de l'état d'un des objets de la particule précédente, et la sortie d'un des objets de la particule précédente. Ces trois mouvements sont respectivement notés *e*, *u* et *s*, et nous noterons *m* le mouvement, tel que  $m \in \mathcal{M} = \{e, u, s\}$ .

- ▷ **Entrée objet** : ce mouvement propose une nouvelle configuration jointe  $\mathbf{X}_t^* = \{\mathbf{X}_t^{n-1}, \mathbf{x}_t^{J_t^{n-1}+1}\}$ , générée par l'ajout d'un nouvel objet  $\mathbf{x}_t^{J_t^{n-1}+1}$  à la précédente configuration jointe  $\mathbf{X}_t^{n-1}$ , où  $J_t^{n-1}$  est le nombre d'objets porté par  $\mathbf{X}_t^{n-1}$ . On donne à l'objet entrant des dimensions *a priori* tirées de lois uniformes sur les intervalles définis par la table 4.1. La figure 4.3 illustre ce mouvement. Le processus de recherche « saute » vers un sous-espace de plus grande dimension.
- ▷ **Sortie objet** : ce mouvement propose une nouvelle configuration jointe excluant l'objet *j* de la particule précédente :  $\mathbf{X}_t^* = \{\mathbf{X}_t^{n-1} \setminus \mathbf{x}_t^{j,n-1}\}$ ,  $j \in \{1, \dots, J_t^{n-1}\}$ , où  $J_t^{n-1}$  est le nombre d'objets dans l'hypothèse  $\mathbf{X}_t^{n-1}$ , et  $\{\mathcal{S} \setminus \mathcal{E}\}$  est l'ensemble  $\mathcal{S}$  privé de l'élément  $\mathcal{E}$ . La figure 4.4 illustre ce mouvement. Le processus de recherche « saute » vers un sous-espace de plus petite dimension.
- ▷ **Mise à jour objet** : dans ce mouvement, on tire au hasard un objet *j* de la particule  $\mathbf{X}_t^{n-1}$  à mettre à jour. Notons  $\mathbf{x}_t^{j,n-1}$  le vecteur décrivant cet objet. On tire au hasard dans la chaîne précédente  $\{\mathbf{X}_{t-1}^n\}_{n=1}^N$  une particule  $\mathbf{X}_{t-1}^r$ ,  $r \in \{1, \dots, N\}$ , qui contient une instance de l'objet *j* à *t* - 1 notée  $\mathbf{x}_{t-1}^{j,r}$ . La figure 4.2 illustre ce mouvement dans le cas d'un modèle dynamique simple où on perturbe la position. Dans le cas d'une commande en vitesse, on tire une proposition de vitesse selon (4.5) :

$$p(\mathbf{v}^* | \mathbf{v}_{t-1}^{j,r}) = \mathcal{N}(\mathbf{v}_{t-1}^{j,r}, \text{diag}([\sigma_m^2, \sigma_a^2])) \quad (4.5)$$

où  $\sigma_m$  et  $\sigma_a$  sont les écarts types respectifs de norme et d'orientation de la vitesse. On applique enfin les équations de la dynamique (4.3) et (4.4) pour mettre à jour la position  $\mathbf{p}_t^{j,n}$  de l'objet *j*. La forme de l'objet est mise à jour selon (4.6) :

$$p(\mathbf{s}^* | \mathbf{s}_{t-1}^{j,r}) = \mathcal{N}(\mathbf{s}_{t-1}^{j,r}, \sigma_s^2 I_3), \quad (4.6)$$

où  $\sigma_s$  est l'écart type de forme.

### 4.3.2 Taux d'acceptation

Le taux d'acceptation requis à chaque itération du *FP MCMC*, se calcule dans le cas général conformément à l'équation (2.10). Cette équation établit la probabilité d'acceptation d'un état proposé  $\mathbf{X}_t^*$ , à partir de l'état à l'itération précédente  $\mathbf{X}_t^{n-1}$ . Compte tenu des divers mouvements *m* mis en œuvre par le *FP RJ-MCMC*, l'évaluation de la loi de proposition  $q(\mathbf{X}_t^* | \mathbf{X}_t^{n-1})$  du nouvel état *via* le mouvement *m* affectant exclusivement l'objet *j*, peut être exprimé par :

$$Q(\mathbf{X}_t^* | \mathbf{X}_t^{n-1}) = Q(m)Q(j|m)Q(\mathbf{x}_t^{j*} | \mathbf{x}_t^{j,n-1}, m) \prod_{s=1, s \neq j}^{J_t^{n-1}} \delta(\mathbf{x}_t^s - \mathbf{x}_t^{s,n-1}), \quad (4.7)$$

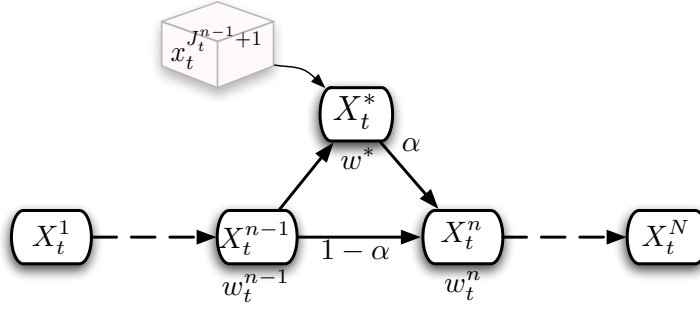


FIGURE 4.3 – Mouvement d'entrée d'un nouvel objet à l'itération  $n$  du filtre *FP MCMC*. La proposition  $\mathbf{X}_t^*$  est construite en ajoutant un nouvel objet à la particule  $\mathbf{X}_t^{n-1}$ . Cette nouvelle configuration  $\mathbf{X}_t^*$  sera acceptée comme prochain maillon de la chaîne  $\mathbf{X}_t^n$  avec la probabilité  $\alpha$ . S'il est rejeté alors on duplique le précédent :  $\mathbf{X}_t^n = \mathbf{X}_t^{n-1}$ .

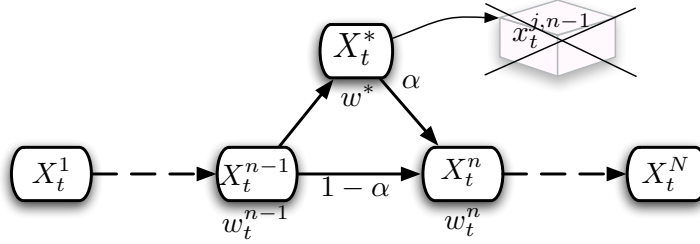


FIGURE 4.4 – Mouvement de sortie à l'itération  $n$ , de l'objet  $\mathbf{x}_t^{j,n-1}$  présent dans le maillon précédent de la chaîne  $\mathbf{X}_t^{n-1}$ . La proposition  $\mathbf{X}_t^*$  est construite en supprimant cet objet de la particule  $\mathbf{X}_t^{n-1}$ . Cette nouvelle configuration  $\mathbf{X}_t^*$  sera acceptée comme prochain maillon de la chaîne  $\mathbf{X}_t^n$  avec la probabilité  $\alpha$ . S'il est rejeté alors on duplique le précédent :  $\mathbf{X}_t^n = \mathbf{X}_t^{n-1}$ .

TABLE 4.1 – Exemples d'intervalles de variation des dimensions *a priori* du modèle géométrique des objets des objets entrants.

catégorie	longueur (m)	largeur (m)	hauteur (m)
piéton	0.25 à 0.4	0.3 à 0.5	1.4 à 1.9
moto	1.5 à 1.8	0.4 à 0.6	1.2 à 1.5
voiture	3.5 à 5.0	1.4 à 2	1.2 à 1.9
camionnette	6.0 à 8.0	2 à 2.3	2.6 à 3.4
poids lourd	12.0 à 18.0	2.5 à 2.6	3.5 à 4.2

où  $Q(m)$  est la probabilité que le mouvement  $m$  soit proposé, évaluation de la loi calculée à l'algorithme 4.4,  $Q(j|m)$  est la probabilité que l'objet  $j$  soit affecté par le mouvement  $m$ , également évaluation de la loi calculée par l'algorithme 4.4. Enfin  $Q(\mathbf{x}_t^{j*}|\mathbf{x}_t^{j,n-1}, m)$  est la probabilité de générer  $\mathbf{x}_t^{j*}$ , état de l'objet  $j$ , en perturbant  $\mathbf{x}_t^{j,n-1}$  par le mouvement  $m$ . Pour respecter l'algorithme *FP RJ-MCMC*, il faut également évaluer la probabilité de proposer le mouvement inverse, soit la probabilité de proposition de  $\mathbf{X}_t^{n-1}$  à partir de  $\mathbf{X}_t^*$ , via le mouvement  $\bar{m}$  perturbant uniquement l'objet  $j$ , où  $\bar{m}$  est le mouvement inverse de  $m$  :

$$Q(\mathbf{X}_t^{n-1}|\mathbf{X}_t^*) = Q(\bar{m})Q(j|\bar{m})Q(\mathbf{x}_t^{j,n-1}|\mathbf{x}_t^{j*}, \bar{m}) \prod_{s=1, s \neq j}^{J_t^{n-1}} \delta(\mathbf{x}_t^s - \mathbf{x}_t^{s,n-1}), \quad (4.8)$$

Le taux d'acceptation défini par l'équation (2.10) peut alors être calculé :

$$\alpha = \min \left( 1, \frac{P(\mathbf{X}_t^*|\mathbf{Z}_{1:t})Q(\mathbf{X}_t^{n-1}|\mathbf{X}_t^*)}{P(\mathbf{X}_t^{n-1}|\mathbf{Z}_{1:t})Q(\mathbf{X}_t^*|\mathbf{X}_t^{n-1})} \right), \quad (4.9)$$

La formule de Bayes et les équations 4.7 et 4.8 permettent de développer le taux d'acceptation :

$$\alpha = \min \left( 1, \frac{w^* P(\mathbf{X}_t^*|\mathbf{Z}_{1:t-1})Q(\bar{m})Q(j|\bar{m})Q(\mathbf{x}_t^{j,n-1}|\mathbf{x}_t^{j*}, \bar{m})}{w_t^{n-1} P(\mathbf{X}_t^{n-1}|\mathbf{Z}_{1:t-1})Q(m)Q(j|m)Q(\mathbf{x}_t^{j*}|\mathbf{x}_t^{j,n-1}, m)} \right), \quad (4.10)$$

avec

$$w^* = P(\mathbf{Z}_t|\mathbf{X}_t^*) \quad (4.11)$$

et

$$w_t^{n-1} = P(\mathbf{Z}_t|\mathbf{X}_t^{n-1}). \quad (4.12)$$

Pour pouvoir proposer l'entrée de  $\mathbf{x}_t^{j_e}$ , nouvel objet d'index  $j_e$ , il faut lui proposer un état initial. On verra que cette initialisation n'est pas facile, et peut être très coûteuse, car on n'a pas d'*a priori* sur cet état. Notons  $p(\mathbf{x}_t^{j_e})$  la loi de probabilité de cet état. On ne peut donc le traiter comme les autres objets, et la loi de prédiction dynamique s'écrit :

$$p(\mathbf{X}_t|\mathbf{X}_{t-1}) = p(\mathbf{x}_t^{j_e}) \prod_{i \in \mathcal{I}_{t-1}} p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i), \quad (4.13)$$

où  $\mathcal{I}_{t-1}$  est l'ensemble des indices des objets présents à l'instant  $t-1$ , et  $p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)$  la loi de probabilité modélisant la dynamique marginalisée sur le sous-espace décrivant l'objet  $i$ . L'approximation échantillonnée (2.8) de cette loi est alors :

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t-1}) \approx p(\mathbf{x}_t^{j_e}) \frac{1}{N} \sum_{\nu=1}^N \prod_{i \in \mathcal{I}_{t-1}^\nu} p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^{i,\nu}), \quad (4.14)$$

où  $\mathcal{I}_{t-1}^\nu$  est l'ensemble des indices des objets candidats présents dans la particule  $\nu$  de la chaîne construite à l'instant  $t-1$ .

▷ Taux d'acceptation d'une mise à jour de l'objet  $j$  : nouvelle configuration jointe  $\mathbf{X}_t^*$ .

$$\alpha_u = \min \left( 1, \frac{w^* \frac{1}{N} \sum_{\nu=1}^N \prod_{i \in \mathcal{I}_{t-1}^\nu} P(\mathbf{x}_t^{i*}|\mathbf{x}_{t-1}^{i,\nu})Q(u)Q(j|u)Q(\mathbf{x}_t^{j,n-1}|\mathbf{x}_t^{j*})}{w_t^{n-1} \frac{1}{N} \sum_{\nu=1}^N \prod_{i \in \mathcal{I}_{t-1}^\nu} P(\mathbf{x}_t^{i,n-1}|\mathbf{x}_{t-1}^{i,\nu})Q(u)Q(j|u)Q(\mathbf{x}_t^{j*}|\mathbf{x}_t^{j,n-1})} \right) \quad (4.15)$$

▷ Taux d'acceptation d'un mouvement d'entrée : nouvelle configuration jointe  $\mathbf{X}_t^* = \{\mathbf{X}_t^{n-1}, \mathbf{x}_t^{j_e}\}$ , ajoutant un nouvel objet  $\mathbf{x}_t^{j_e}$  d'indice  $j_e$ , à la précédente configuration jointe  $\mathbf{X}_t^{n-1}$  :

$$\alpha_e = \min \left( 1, \frac{w^* P(\mathbf{x}_t^{j_e}) \frac{1}{N} \sum_{\nu=1}^N \prod_{i \in \mathcal{I}_{t-1}^\nu} P(\mathbf{x}_t^{i*}|\mathbf{x}_{t-1}^{i,\nu})Q(s)Q(j_e|s)}{w_t^{n-1} \frac{1}{N} \sum_{\nu=1}^N \prod_{i \in \mathcal{I}_{t-1}^\nu} P(\mathbf{x}_t^{i,n-1}|\mathbf{x}_{t-1}^{i,\nu})Q(e)Q(\mathbf{x}_t^{j_e})} \right), \quad (4.16)$$

Le plus naturel est de prendre une loi de proposition du nouvel objet  $q(\mathbf{x}_t^{j_e})$  identique à la loi *a priori*  $p(\mathbf{x}_t^{j_e})$ , donc  $Q(\mathbf{x}_t^{j_e}) = P(\mathbf{x}_t^{j_e})$ . Par ailleurs,  $P(\mathbf{x}_t^{i*}|\mathbf{x}_{t-1}^{i,\nu}) = P(\mathbf{x}_t^{i,n-1}|\mathbf{x}_{t-1}^{i,\nu})$ , car  $\mathbf{x}_t^{i*} = \mathbf{x}_t^{i,n-1} \forall i \in \mathcal{I}_{t-1}^\nu$ . Le calcul du taux d'acceptation se simplifie ainsi :

$$\alpha_e = \min \left( 1, \frac{w^* Q(s) Q(j_e|s)}{w_t^{n-1} Q(e)} \right) \quad (4.17)$$

- ▷ Taux d'acceptation d'un mouvement de sortie de l'objet  $j$  : nouvelle configuration jointe  $\mathbf{X}_t^* = \{\mathbf{X}_t^{n-1} \setminus \mathbf{x}_t^{j,n-1}\}$ ,  $j \in \{1, \dots, J_t^{n-1}\}$ , où  $J_t^{n-1}$  est le nombre d'objets dans l'hypothèse  $\mathbf{X}_t^{n-1}$ , et  $\{\mathcal{S} \setminus \mathcal{E}\}$  est l'ensemble  $\mathcal{S}$  privé de l'élément  $\mathcal{E}$  :

$$\alpha_s = \min \left( 1, \frac{w^* \frac{1}{N} \sum_{\nu=1}^N \prod_{i \in \{\mathcal{I}_{t-1}^\nu \setminus j\}} P(\mathbf{x}_t^{i*}|\mathbf{x}_{t-1}^{i,\nu}) Q(e) Q(\mathbf{x}_t^{j,n-1})}{w_t^{n-1} P(\mathbf{x}_t^{j,n-1}) \frac{1}{N} \sum_{\nu=1}^N \prod_{i \in \{\mathcal{I}_{t-1}^\nu \setminus j\}} P(\mathbf{x}_t^{i,n-1}|\mathbf{x}_{t-1}^{i,\nu}) Q(s) Q(j|s)} \right) \quad (4.18)$$

Le terme  $Q(\mathbf{x}_t^{j,n-1})$  au numérateur est l'évaluation de la loi de proposition de l'état  $\mathbf{x}_t^{j,n-1}$  pour l'objet  $j$  dans le mouvement inverse (son entrée). Ici aussi, le plus naturel est d'utiliser pour loi de proposition de cette entrée la loi *a priori*  $p(\mathbf{x}_t^{j_e})$ , donc  $Q(\mathbf{x}_t^{j,n-1}) = P(\mathbf{x}_t^{j,n-1})$ . Par ailleurs,  $P(\mathbf{x}_t^{i*}|\mathbf{x}_{t-1}^{i,\nu}) = P(\mathbf{x}_t^{i,n-1}|\mathbf{x}_{t-1}^{i,\nu})$ , car  $\mathbf{x}_t^{i*} = \mathbf{x}_t^{i,n-1} \forall i \in \{\mathcal{I}_{t-1}^\nu \setminus j\}$ . Le calcul du taux d'acceptation se simplifie ainsi :

$$\alpha_s = \min \left( 1, \frac{w^* Q(e)}{w_t^{n-1} Q(s) Q(j|s)} \right) \quad (4.19)$$

### 4.3.3 Gestion aveugle des mouvements

Une implémentation basique du filtre *FP RJ-MCMC* consiste à échantillonner à chaque instant  $t$  la loi de probabilité *a posteriori* de l'état joint multi-objets  $\mathbf{X}_t$  en explorant l'espace des configurations jointes  $\mathcal{X}$  au moyen de propositions construites à partir des particules de la chaîne à l'instant précédent  $t - 1$ . Dans une optique temps réel, nous avons conçu un filtre *FP RJ-MCMC* à nombre de particules  $N$  constant, c'est à dire le maximum compatible avec la fréquence de traitement et les possibilités de la machine. Le nombre d'objets étant variable et leurs mouvements marginalisés, il convient de répartir équitablement ces  $N$  itérations disponibles entre mises à jour et les sorties des objets présents et l'entrée d'un nouvel objet. Dans ce cas les lois  $q(m)$ ,  $m \in \mathcal{M} = \{e, u, s\}$ , et  $q(j|m)$ ,  $j$  étant l'objet bougé, dépendent en fait du nombre d'objets portés par la particule à l'itération précédente. Elles devraient donc être notées respectivement  $q(m|\mathbf{X}_t^{n-1})$  et  $q(j|\mathbf{X}_t^{n-1}, m)$ . L'algorithme 4.2 explicite les calculs de ces lois, intervenant dans les calculs des taux d'acceptation 4.17 et 4.19. Elles sont par contre indépendantes de l'observation courante  $\mathbf{Z}_t$ .

### 4.3.4 Nécessité de guider les mouvements

Une expérimentation de la méthode avec  $q(m)$ ,  $q(j|m)$ , et  $q(\mathbf{x}^{j_e})$  définies par l'algorithme 4.2, montre qu'on va consommer beaucoup d'itérations pour initialiser les objets entrants. En particulier si on n'a pas d'*a priori* sur l'état d'un objet entrant, il faudrait prendre une loi de proposition de position  $q(\mathbf{p}^{j_e}) = \mathcal{U}_{\mathcal{T}}$ , où  $\mathcal{U}_{\mathcal{T}}$  est une loi uniforme sur l'espace de traque  $\mathcal{T}$ . En surveillance routière, cet espace couvre une surface d'environ  $5000m^2$ . Une proposition d'objet entrant non guidée par la mesure est donc impossible. Il est alors indispensable d'adapter ces lois au contexte à chaque itération, c'est à dire à la fois à l'observation courante  $\mathbf{Z}_t$  et à la particule précédente  $\mathbf{X}_t^{n-1}$ . On proposera par exemple davantage d'entrées à l'itération  $n$  lorsqu'un amas de pixels classifiés comme *forme* ne coïncide avec aucun des objets portés par la particule  $\mathbf{X}_t^{n-1}$ . Ces calculs faisant intervenir l'observation courante  $\mathbf{Z}_t$ , ils seront explicités en section 4.5 pour les entrées, et en section 4.6 pour les sorties. Enfin l'algorithme 4.4 redéfinira ces lois de probabilités guidées par l'observation.



**Algorithme 4.2** Gestion aveugle des propositions de mouvements du *FP RJ-MCMC***Entrées :**

- fréquence de proposition d'une entrée  $f_e$ , relative aux mises à jour.
- fréquence de proposition d'une sortie  $f_s$ , relative aux mises à jour.
- ensemble des types de mouvements  $m \in \mathcal{M} = \{e, u, s\}$  où  $e$  représente un mouvement d'entrée,  $u$  représente la mise à jour de l'état d'un des objets de  $\mathbf{X}_t^{n-1}$ ,  $s$  représente la sortie d'un des objets de  $\mathbf{X}_t^{n-1}$ .
- particule précédente  $\mathbf{X}_t^{n-1}$  contenant  $J_t^{n-1}$  objets.
- régions d'entrée d'objets dans l'espace de traque 3D.

**Calcul de la loi de mouvements  $q(m|\mathbf{X}_t^{n-1})$  :**

- entrées :  $Q(e|\mathbf{X}_t^{n-1}) = f_e$
- mises à jour :  $Q(u|\mathbf{X}_t^{n-1}) = J_t^{n-1}$
- sorties :  $Q(s|\mathbf{X}_t^{n-1}) = f_s J_t^{n-1}$

**Calcul de la loi de choix de l'objet affecté  $q(j|\mathbf{X}_t^{n-1}, m)$  :**

- choix objet mis à jour :  $Q(j|\mathbf{X}_t^{n-1}, u) = 1/J_t^{n-1}, \forall j \in \{1, \dots, J_t^{n-1}\}$
- choix objet sorti :  $Q(j|\mathbf{X}_t^{n-1}, s) = 1/J_t^{n-1}, \forall j \in \{1, \dots, J_t^{n-1}\}$

**Normalisation :**

- normaliser :  $Q(m|\mathbf{X}_t^{n-1}) = Q(m|\mathbf{X}_t^{n-1})(f_e + J_t^{n-1}(1 + f_s))^{-1}$
- calculer  $\Sigma = Q(e|\mathbf{X}_t^{n-1}) + \sum_{m \neq e} \sum_{j=1}^{J_t^{n-1}} Q(m|\mathbf{X}_t^{n-1})Q(j|\mathbf{X}_t^{n-1}, m)$
- normaliser :  $Q(j|\mathbf{X}_t^{n-1}, m) = Q(j|\mathbf{X}_t^{n-1}, m)\Sigma^{-1}$

**Sorties :**

- $q(m) = q(m|\mathbf{X}_t^{n-1})$ ,  $m \in \mathcal{M} = \{e, u, s\}$
- $q(j|m) = q(j|\mathbf{X}_t^{n-1}, m)$ ,  $m \in \mathcal{M} = \{e, u, s\}$ ,  $j \in \{1, \dots, J_t^{n-1}\}$
- $q(\mathbf{x}^{j_e})$ , loi de probabilité de l'état *a priori* d'un objet entrant.

**4.3.5 Mise en œuvre du *FP RJ-MCMC***

L'algorithme 4.3 détaille le processus de suivi d'un nombre variable d'objets. C'est la généralisation de l'algorithme 4.1, dans lequel seules les mises à jour d'objets étaient possibles. Pour suivre un nombre variable d'objets, l'algorithme 4.3, propose à chaque itération soit l'entrée d'un nouvel objet, soit la sortie d'un des objets présents à l'itération précédente, soit la mise à jour d'un des objets présents à l'itération précédente. Dans tous les cas, ce mouvement, tiré selon la loi  $q(m)$ , perturbe exclusivement l'objet  $j$ , tiré selon la loi  $q(j|m)$ , les autres objets restant inchangés.

**Estimation de l'état.** Dans le cas du suivi d'un nombre fixe d'objets par *FP MCMC*, on a vu en section 4.2.4, qu'il est avantageux de choisir comme estimation de l'état  $\mathbf{X}_t$  la particule la plus recopiée, approximation du *Maximum A Posteriori*. Dans le cas du suivi d'un nombre variable d'objets par *FP RJ-MCMC*, un argument supplémentaire vient renforcer ce choix : la présence de particules de dimensions différentes car portant des nombres d'objets différents, rend délicat un calcul de moyenne. Cet argument prend d'autant plus de poids, que des variables aléatoires discrètes vont être introduites dans les sections suivantes, telles que la présence du soleil ou la catégorie d'un objet, sur lesquelles la moyenne prend peu de sens.

**4.4 Modèle d'observation**

Les applications de suivi multi-objets que nous visons présentent les contraintes suivantes : nombre variable d'objets, fortes occultations inter-objets, fortes variation de taille apparente des objets dues à la perspective, grandes disparité de dimensions réelles des objets suivis (moto contre semi-remorque dans le trafic routier), fortes variations d'éclairement. Nous décrivons dans cette section un modèle d'observation conçu pour répondre à ces contraintes. Nous proposons une fonction de vraisemblance basée sur une carte de segmentation binaire *fond / forme*. Pour gérer les occultations, nous avons choisi d'utiliser une observation globale bas niveau de l'image : une carte binaire *fond / forme*. Il n'y a donc pas de détecteur en amont du filtre, et on peut



**Algorithme 4.3** Suivi d'un Nombre Variable d'Objets par *FP RJ-MCMC***Entrées :**

- nuage de particules approximant la loi de probabilité *a posteriori* de l'état joint multi-objets  $\mathbf{X}_{t-1}$  à l'instant  $t - 1$  :  $\{\mathbf{X}_{t-1}^n\}_{n=1}^N$ , tel que  $\mathbf{X}_{t-1}^n = [\mathbf{x}_{t-1}^{j,n}]_{j=1}^{J_{t-1}^n}$ , où  $\mathbf{x}_{t-1}^{j,n}$  est le descripteur de l'objet  $j$ , de dimension  $d$ ,  $\forall j \in \{1, \dots, J_{t-1}^n\}$ .  $J_{t-1}^n$  est le nombre d'objets décrits par la particule  $n$ .
- lois de probabilité de prédiction marginales  $p(\mathbf{x}_t^j | \mathbf{x}_{t-1}^j)$ , modélisant la dynamique de l'objet  $j$ .
- modèle de vraisemblance d'observation  $p(\mathbf{Z} | \mathbf{X})$  pour le calcul de  $w(\mathbf{X})$ , poids de la particule  $\mathbf{X}$ .
- observation  $\mathbf{Z}_t$  à l'instant  $t$ .

**Initialisation de la chaîne :**

- Tirer aléatoirement une particule de la chaîne précédente :  $\mathbf{X}_{t-1}^r$ ,  $r \in \{1, \dots, N\}$ , contenant  $I_{t-1}^r$  objets.
- Générer une particule initiale  $\mathbf{X}_t^0 = [\mathbf{x}_t^{j,0}]_{j=1}^{I_{t-1}^r}$ , où  $\mathbf{x}_t^{j,0} \sim p(\mathbf{x}_t^j | \mathbf{x}_{t-1}^{j,r}) \forall j \in \{1, \dots, I_{t-1}^r\}$ .
- Évaluer son poids  $w_t^0$  selon l'équation (4.11).

**pour**  $n = 1$  à  $N + N_B$  **faire**

- calculer les lois  $q(m)$ ,  $q(j|m)$ , et  $q(\mathbf{x}^{je})$ ,  $m \in \mathcal{M}$  par l'algorithme 4.2 ou par 4.4 .
- Tirer le mouvement  $m$  à effectuer  $m \sim q(m)$

**si**  $m == u$  (mise à jour objet) **alors**

- Tirer l'indice de l'objet à bouger :  $j \sim q(j|u)$
- Propagation : tirer aléatoirement une particule de la chaîne précédente :  $\mathbf{X}_{t-1}^r$ ,  $r \in \{1, \dots, N\}$ , et extraire l'instance d'objet  $j$  qu'elle contient :  $\mathbf{x}_{t-1}^{j,r}$
- Prédiction : générer  $\mathbf{x}_t^{j*} \sim q(\mathbf{x}_t^j | \mathbf{x}_{t-1}^{j,r})$ , proposition d'objet  $j$  tirée de la loi marginale dynamique.
- Générer une particule prédite  $\mathbf{X}_t^*$ , en remplaçant  $\mathbf{x}_t^{j,n-1}$ , instance d'objet  $j$  de  $\mathbf{X}_t^{n-1}$ , par  $\mathbf{x}_t^{j*}$ .

**sinon si**  $m == s$  (sortie objet) **alors**

- Tirer l'indice de l'objet à sortir :  $j \sim q(j|s)$
- Générer une particule prédite  $\mathbf{X}_t^*$ , en supprimant  $\mathbf{x}_t^{j,n-1}$ , l'objet  $j$  de  $\mathbf{X}_t^{n-1}$ .

**sinon si**  $m == e$  (entrée objet) **alors**

- Tirer un état pour cet objet entrant :  $\mathbf{x}^* \sim q(\mathbf{x}^{je})$
- Générer une particule prédite  $\mathbf{X}_t^*$ , en ajoutant cet objet  $\mathbf{x}^*$  à  $\mathbf{X}_t^{n-1}$ .

**fin si**

- Évaluation : calculer son poids  $w^*$  selon l'équation (4.11).
- Calcul du taux d'acceptation  $\alpha_m$ , selon l'une des équations (4.15), (4.17), ou (4.19), selon  $m$ .
- Ajout du  $i^{\text{ième}}$  élément de la chaîne :  $\mathbf{X}_t^i = \mathbf{X}_t^*$  et  $w_t^n = w^*$  avec la probabilité  $\alpha_m$ , ou duplication du précédent  $\mathbf{X}_t^i = \mathbf{X}_t^{i-1}$  et  $w_t^n = w_t^{n-1}$  avec la probabilité  $1 - \alpha_m$ .

**fin pour**

**Burn-in** : supprimer les  $N_B$  premiers échantillons de la chaîne.

**Sortie** : nuage de particules approximant la loi de probabilité *a posteriori* de  $\mathbf{X}_t$  à l'instant  $t$  :

$$\{\mathbf{X}_t^n\}_{n=N_B+1, \dots, N_B+N}$$

considérer que l'étape de détection d'objet est intégrée au filtre. L'inconvénient de cette approche est que les occultations rendent cette observation globale impossible à partitionner en scores individuels de chaque objet, aussi l'exploration se fait sur l'espace d'état joint, avec une observation jointe.

Par ailleurs, les caméras de surveillance, notamment routière, ne sont pas toujours placées en un point très élevé, ce qui cause de fortes occultations entre objets. En conséquence, la fonction de vraisemblance doit comparer équitablement les objets, qu'ils soient occultés ou non. De plus, la grande étendue de mesure (plusieurs centaines de mètres) provoque des variations considérable de la surface apparente des objets du fait de la projection. Le modèle d'observation doit donc être compatible avec ces conditions. Dans cette section, nous noterons  $\mathbf{X}$  une particule décrivant une configuration jointe multi-objets, et pour laquelle la vraisemblance de l'observation jointe doit être évaluée. L'observation est réalisée par un ensemble de  $C$  caméras, et on note cette observation :

$$\mathbf{Z} = \{\mathbf{I}_B(g, c)\}_{c=1}^C, \quad (4.20)$$

où  $\mathbf{I}_B(g, c)$  est l'image brute envoyée par la caméra  $c$ .

#### 4.4.1 Image masque de particule

Pour chaque particule  $\mathbf{X}$ , on calcule une « image masque de particule », image binaire définie comme l'union des projections des objets de  $\mathbf{X}$  dans le plan image de la caméra, selon l'équation (4.21) :

$$\mathbf{I}_M(\mathbf{X}, g, c) = \bigcup_{i=1}^I \mathcal{I}_{j,c}, \forall g \in \{1, \dots, G\}, \forall c \in \{1, \dots, C\} \quad (4.21)$$

où  $\mathcal{I}_{j,c}$  est la région englobée par l'enveloppe convexe de la projection dans le plan image de la caméra  $c$ , des sommets du polyèdre 3D modélisant l'objet  $j$ . Cette image est du type de la figure 4.5-(c), où le pixel  $g$  vaut 1 s'il est à l'intérieur d'au moins une des enveloppes convexes des projections des parallélépipèdes d'objets de la particule dans l'image, sinon 0.

#### 4.4.2 Segmentation *fond / forme*

A partir de l'image courante (figure 4.5-a), et d'un modèle de *fond* (figure 4.5-b), une image binaire d'avant plan  $\mathbf{I}_F(\mathbf{Z}, g, c)$  est calculée (figure 4.5-d), par l'algorithme  $\Sigma - \Delta$  (MR04). Cet algorithme calcule efficacement en ligne une approximation de la médiane et de la variance temporelles de l'image de *fond*. Les deux sont remises à jour à chaque image, ce qui permet de faire face aux changements d'illumination et aux bruits pour un faible coût de calcul.

#### 4.4.3 Vraisemblance d'observation naïve

La fonction de vraisemblance est basée sur la segmentation binaire *fond / forme*, connaissant la particule proposée  $\mathbf{X}$  et l'observation  $\mathbf{Z}$ . On calcule d'une part une image intersection :

$$\mathbf{I}_\cap(\mathbf{X}, \mathbf{Z}, g, c) = \begin{cases} 1 & \text{si } \mathbf{I}_F(\mathbf{Z}, g, c) \text{ et } \mathbf{I}_M(\mathbf{X}, g, c) \\ 0 & \text{sinon} \end{cases} \quad \forall g \in \{1, \dots, G\}, \forall c \in \{1, \dots, C\} \quad (4.22)$$

Et d'autre part une image réunion :

$$\mathbf{I}_\cup(\mathbf{X}, \mathbf{Z}, g, c) = \begin{cases} 1 & \text{si } \mathbf{I}_F(\mathbf{Z}, g, c) \text{ ou } \mathbf{I}_M(\mathbf{X}, g, c) \\ 0 & \text{sinon} \end{cases} \quad \forall g \in \{1, \dots, G\}, \forall c \in \{1, \dots, C\} \quad (4.23)$$

De ces images réunion et intersection, on peut tirer une fonction de vraisemblance définie par le ratio :

$$w(\mathbf{X}, \mathbf{Z}, c) = p(\mathbf{I}_B(g, c) | \mathbf{X}) = \frac{\sum_{g \in \{1, \dots, G\}} \mathbf{I}_\cap(\mathbf{X}, \mathbf{Z}, g, c)}{\sum_{g \in \{1, \dots, G\}} \mathbf{I}_\cup(\mathbf{X}, \mathbf{Z}, g, c)} \quad (4.24)$$

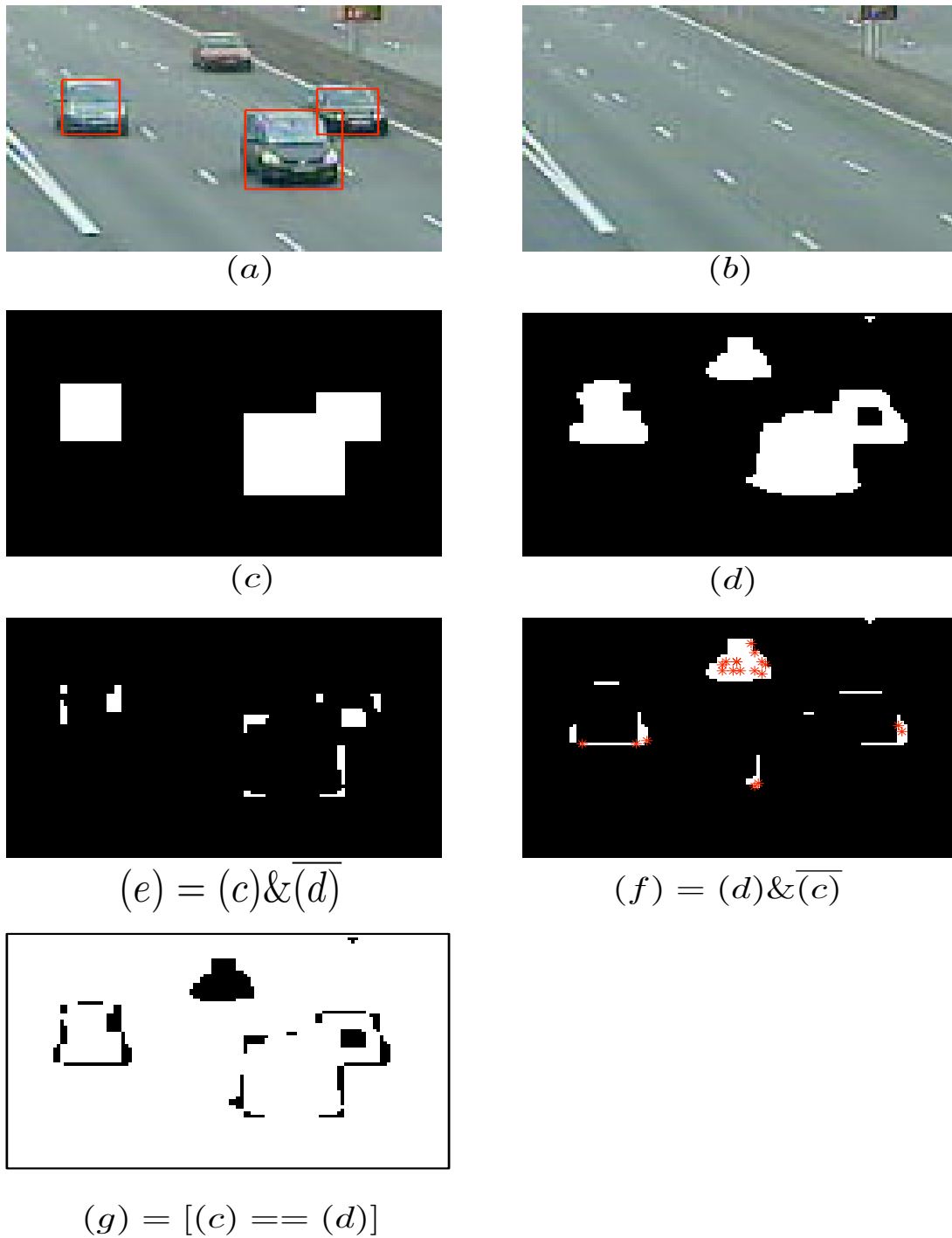


FIGURE 4.5 – Segmentation *fond / forme* et images des résidus en mono-vision. (a) : images couleur avec les objets en rouge. Pour des raisons de lisibilité, les projections des objets sont approximées par des rectangles. (b) : modèle de *fond*. (c) : image binaire hypothèse  $I_M(X)$ . (d) : image binaire d'avant-plan  $I_F(Z)$ . (e) : image binaire de « faux avant plan », pixels couverts par la projection d'au moins un objet, mais classés comme arrière-plan. (f) : image binaire de « faux arrière plan », pixels non couverts par la projection d'aucun objet hypothèse, mais classifiés comme avant-plan. cette image sert au guidage des propositions d'entrée d'un nouvel objet : les points rouges sont des échantillons tirés parmi les amas de « faux arrière plan », puis subissent une homographie vers le plan horizontal supposé contenir les centres de gravité des objets, afin de proposer la position d'un nouvel objet. (g) : image binaire de similarité  $I_S(X, Z)$  entre l'image binaire hypothèse  $I_M(X)$  représentée en (c) et l'image binaire d'avant-plan  $I_F(Z)$  représentée en (d).

Cette fonction serait d'autant plus séduisante, que le calcul peut être réduit à la région contenue dans  $I_U(\mathbf{X}, \mathbf{Z}, g, c)$ , ce qui évite de parcourir toute l'image. Elle est adaptée lorsque les effets de perspective sont négligeables, dans le cas par exemple d'un suivi en 2D avec caméra au plafond avec axe optique vertical. Malheureusement, dans le cas général de nos applications, notamment en suivi de trafic, la grande étendue de mesure (plusieurs centaines de mètres) provoque des variations considérables de la surface apparente des objets du fait de la perspective. En conséquence, la fonction de vraisemblance définie par (4.24), est beaucoup plus discriminante pour les objets proches que pour les objets lointains. Cet effet est illustré par la figure 4.6, où on a proposé à chaque objet candidat un mouvement horizontal de la moitié de sa largeur. Les nombres placés au dessus de chaque objet candidat, indiquent le taux d'acceptation calculé avec la formule. Tandis que 71% des mouvements proposés à l'objet proche seraient acceptés, la quasi-totalité des mouvements proposés à l'objet lointain (96%) seraient acceptés, c'est à dire que l'on n'est quasiment pas discriminant sur l'objet lointain. Ce constat est généralisé par la figure 4.7, dont la courbe rouge met en évidence la chute du taux d'acceptation lorsque l'objet se rapproche de la caméra (*i.e.* lorsque la surface de la projection de l'objet augmente).

#### 4.4.4 Vraisemblance relative à la distance objet-caméra

Le calcul de la vraisemblance présenté dans cette section permet de surmonter cette difficulté. Il est fortement dépendant de la surface de l'objet, de façon à assurer à l'algorithme de Metropolis-Hastings un taux d'acceptation constant, indépendamment de la surface de l'objet. Il a été proposé dans (BC08a). Ce calcul est rendu possible par le fait que les algorithmes 4.1 et 4.3 proposent de bouger un seul objet candidat à la fois. On calcule d'abord une image de similarité  $I_S(\mathbf{X}, \mathbf{Z}, g, c)$ , selon (4.25) :

$$I_S(\mathbf{X}, \mathbf{Z}, g, c) = \begin{cases} 1 & \text{si } \mathbf{I}_F(\mathbf{Z}, g, c) = \mathbf{I}_M(\mathbf{X}, g, c) \\ 0 & \text{sinon} \end{cases} \quad \forall g \in \{1, \dots, G\}, \forall c \in \{1, \dots, C\} \quad (4.25)$$

Puis on calcule une variable notée  $\beta_{j,c} \forall c \in \{1, \dots, C\}$ , telle qu'une perte  $k_S$  de la couverture de l'objet cible par l'objet candidat, conduise à un taux d'acceptation cible  $\alpha_t$ . Notons  $S$  la surface de l'image binaire d'avant plan  $\mathbf{I}_F(\mathbf{Z}, g, c)$ ,  $S_{j,c}$  la surface contenue dans l'enveloppe convexe du polygone projeté de l'objet  $j$  dans l'image de la caméra  $c$ . Pour nos expérimentations, nous avons choisi  $k_S = 0.1$  et  $\alpha_t = 0.5$ , c'est à dire qu'une proposition de mouvement d'un objet candidat, conduisant à une perte de couverture de l'objet cible de 10% serait acceptée avec une probabilité  $\alpha_t = 0.5$ , ce qui correspond à l'équiprobabilité de l'acceptation et du refus d'une telle proposition. La vraisemblance est calculée ainsi :

$$w(\mathbf{X}, \mathbf{Z}, c, j) = p(\mathbf{I}_F(\mathbf{Z}, g, c) | \mathbf{X}, j) = \left( \frac{1}{S} \sum_{g \in \{1, \dots, G\}} I_S(\mathbf{X}, \mathbf{Z}, g, c) \right)^{\beta_{j,c}} \quad \forall c \in \{1, \dots, C\}. \quad (4.26)$$

Dans le cas où le masque de l'objet candidat  $\mathbf{I}_M(\mathbf{X}, g, c)$  correspond parfaitement à  $\mathbf{I}_F(\mathbf{X}, g, c)$ , ce calcul donne :

$$w(\mathbf{X}, \mathbf{Z}, c, j) = 1, \quad (4.27)$$

et dans le cas où ils n'ont aucune intersection :

$$w(\mathbf{X}, \mathbf{Z}, c, j) = \left( \frac{S - 2S_{j,c}}{S} \right)^{\beta_{j,c}} \quad \forall c \in \{1, \dots, C\} \quad (4.28)$$

La variable  $\beta_{j,c}$  est calculée pour chaque projection de l'objet  $j$  dans chaque caméra  $c$ , par :

$$\beta_{j,c} = \frac{\log(\alpha_t)}{\log(1 - 2k_S \frac{S_{j,c}}{S})} \quad (4.29)$$

Enfin, les vraisemblances de toutes les observations sont fusionnées par l'équation (4.30), qui repose sur l'hypothèse (assez grossière) d'indépendance de ces observations. Cette méthode simpliste de fusion des observations

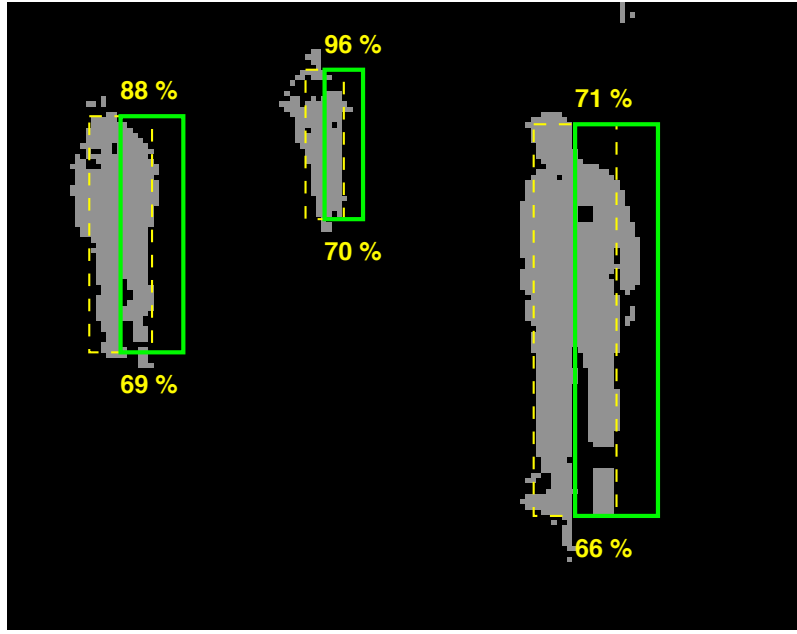


FIGURE 4.6 – Image binaire *fond* (en noir) et *forme* en gris, et la projection d'une hypothèse de configuration jointe multi-objets (rectangles). En pointillés jaunes : hypothèse initiale. En trait continu vert : mouvements proposés sur chaque objet : une translation horizontale d'amplitude égale à la demi-largeur de l'objet. Les taux d'acceptation calculés avec la vraisemblance naïve, sont indiqués au dessus du rectangle. Les taux d'acceptation calculés avec la vraisemblance relative à la distance objet-caméra, sont indiqués au dessous du rectangle.

mériterait d'être revue. Il conviendrait par exemple de pondérer chaque  $w(\mathbf{X}, \mathbf{Z}, c, j)$  par l'observabilité de l'objet  $j$  dans chaque caméra  $c$ .

$$w(\mathbf{X}, \mathbf{Z}, j) = \prod_{c=1}^C w(\mathbf{X}, \mathbf{Z}, c, j) \quad (4.30)$$

A chaque itération, on peut finalement calculer le taux d'acceptation conformément aux équations (4.15), (4.17), et (4.19), **les deux** vraisemblances  $w$  et  $w_t^{n-1}$  étant calculées selon les exposants  $\beta_{j,c}$  relatifs au mouvement de l'objet  $j$ . Pour éviter de calculer deux vraisemblances à chaque itération, on conserve le terme  $\left(\frac{1}{S} \sum_{g \in \{1, \dots, G\}} \mathbf{I}_S(\mathbf{X}, \mathbf{Z}, g, c)\right)$  pour l'itération suivante, et il ne restera plus qu'à l'élever à l'exposant  $\beta_{j,c}$  dès que l'objet bougé  $j$  sera connu.

Les valeurs numériques données en dessous des objets sur la figure 4.6, montrent que les taux d'acceptation obtenus par cette méthode ne dépendent plus de la distance objet-caméra. Ce constat est généralisé par la figure 4.7, dont la courbe noire montre l'indépendance du taux d'acceptation à la surface de la projection de l'objet. Les disparités résiduelles dans ces valeurs sont dues à la correspondance imparfaite entre les silhouettes humaines cibles, et les candidats rectangulaires. De plus, cette méthode nous donne l'opportunité d'asservir le taux d'acceptation avec lequel fonctionne l'algorithme de Metropolis-Hastings, et de l'ajuster à une valeur maximisant son efficacité.

#### 4.4.5 Optimisation du calcul de la vraisemblance

Le gros inconvénient de la vraisemblance présentée dans la section 4.4.4, est que son calcul s'étend sur toute l'image, comme le montre l'équation 4.26, ce qui demande des calculs lourds, pénalisant les performances temps réel. Il y a donc de la marge d'optimisation, car on ne bouge qu'un objet à la fois à chaque itération.

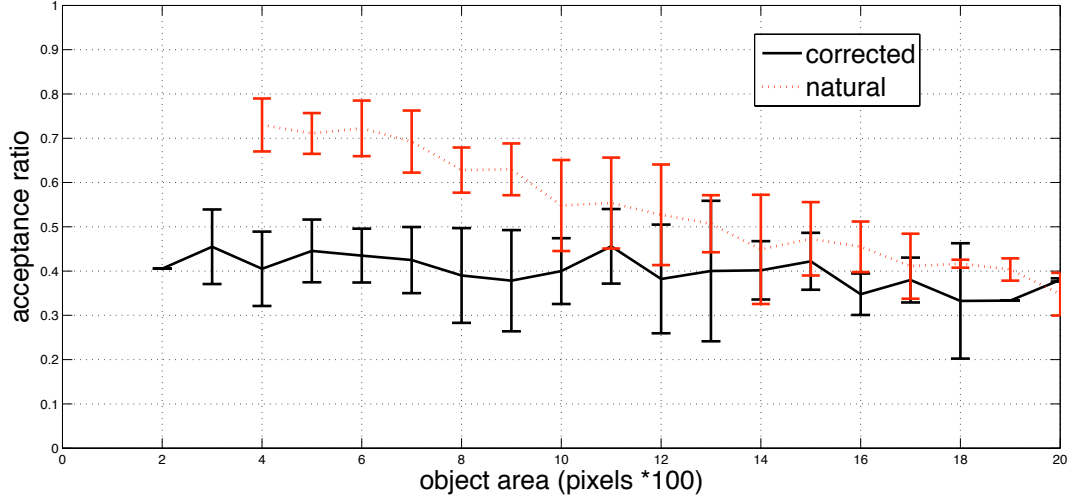


FIGURE 4.7 – Taux d'acceptation moyen calculé dans l'algorithme de Metropolis-Hastings, en fonction de la surface apparente de l'objet. La demi-hauteur des segments verticaux est égale à un écart-type des mesures. Pointillés rouges : vraisemblance naïve. Ligne noire continue : vraisemblance relative à la distance objet-caméra.

Cette marge de progrès, non encore implémentée, consistera à ne calculer la vraisemblance sur toute l'image que pour la première particule de la chaîne, conformément à l'équation 4.26, puis à incrémenter à chaque itération le terme somme de l'image de similarité de la façon suivante :

$$w(\mathbf{X}_t^n, \mathbf{Z}_t, c, j) = \left( \frac{1}{S} \sum_{g \in \mathcal{P}_t^n} \mathbf{I}_S(\mathbf{X}_t^n, \mathbf{Z}_t, g, c) + \frac{1}{S} \sum_{g \in \overline{\mathcal{P}_t^n}} \mathbf{I}_S(\mathbf{X}_t^n, \mathbf{Z}_t, g, c) \right)^{\beta_{j,c}}, \quad (4.31)$$

où  $\mathcal{P}_t^n$  est la région de l'image modifiée par le mouvement de l'objet  $j$  à l'itération  $n$ , c'est à dire l'union des projection de l'objet  $j$  à l'itération  $n - 1$ , et de l'objet  $j$  à l'itération  $n$ . L'équation 4.31 peut aussi s'écrire en fonction de l'état à l'itération précédente :

$$w(\mathbf{X}_t^n, \mathbf{Z}_t, c, j) = \left( \frac{1}{S} \sum_{g \in \mathcal{P}_t^n} \mathbf{I}_S(\mathbf{X}_t^n, \mathbf{Z}_t, g, c) + \frac{1}{S} \sum_{g \in \overline{\mathcal{P}_t^n}} \mathbf{I}_S(\mathbf{X}_t^{n-1}, \mathbf{Z}_t, g, c) \right)^{\beta_{j,c}}, \quad (4.32)$$

$$w(\mathbf{X}_t^n, \mathbf{Z}_t, c, j) = \left( \frac{1}{S} \sum_{g \in \mathcal{P}_t^n} \mathbf{I}_S(\mathbf{X}_t^n, \mathbf{Z}_t, g, c) - \frac{1}{S} \sum_{g \in \mathcal{P}_t^n} \mathbf{I}_S(\mathbf{X}_t^{n-1}, \mathbf{Z}_t, g, c) + \underbrace{\frac{1}{S} \sum_{g \in \{1, \dots, G\}} \mathbf{I}_S(\mathbf{X}_t^{n-1}, \mathbf{Z}_t, g, c)}_{w(\mathbf{X}_t^{n-1}, \mathbf{Z}_t, c, j)^{1/\beta_{j,c}}} \right)^{\beta_{j,c}}, \quad (4.33)$$

## 4.5 Mouvements d'entrée guidés par l'observation

Pour initialiser les objets entrant dans l'espace de traque, deux options se présentent :

- ▷ On externalise le processus d'entrée, c'est à dire qu'un détecteur d'objet alimente le filtre, et celui-ci n'assure que son rôle de filtre, c'est à dire assure la cohérence temporelle des états du système. C'est l'approche choisie par Khan *et al.* (KBD05).

▷ On intègre au filtre la fonction détection d'objet et le processus d'entrée.

Nous avons choisi la deuxième approche, pour éviter les défaillances que rencontrent les détecteurs en cas d'occultation. Ce choix pose au filtre le problème de l'initialisation des objets entrants. Ce problème peut paraître marginal du point de vue du filtre. Il ne l'est pas, à cause des dimensions de l'espace de traque requis dans les applications de suivi d'objet : jusqu'à quelques centaines de mètres. Dans tous les cas, l'espace de traque mesure de 10 à 100 fois la taille des objets suivis. Soit, ramené en surface, un rapport compris entre  $10^2$  et  $10^4$ . Dans ces conditions et en l'absence d'*a priori* sur la localisation des objets entrants, il est illusoire de confier cette initialisation à une méthode de Monte-Carlo, car un trop grand nombre d'hypothèses est requis par ce seul processus, si bien qu'il faudrait consacrer la quasi-totalité des itérations à tester si un nouvel objet entre ou non. Pour que le processus d'initialisation d'objets entrants ne pénalise pas trop le fonctionnement du *FP RJ-MCMC*, il faut pouvoir le guider, c'est à dire lui fournir à chaque itération :

- ▷  $q(e) = q(e|\mathbf{X}_t^{n-1}, \mathbf{Z}_t)$ , loi de probabilité de proposition d'une entrée en fonction de l'observation courante  $\mathbf{Z}_t$ , et de la particule à l'itération précédente  $\mathbf{X}_t^{n-1}$ . L'évaluation de cette loi est requise pour le calcul des taux d'acceptation définis par les équations (4.17) et (4.19)
- ▷  $q(\mathbf{x}_t^{j_e}) = q(\mathbf{x}_t^{j_e}|\mathbf{X}_t^{n-1}, \mathbf{Z}_t)$ , loi définissant l'état initial d'un objet entrant où  $\mathbf{x}_t^{j_e}$  est l'état de l'objet entrant défini selon l'équation (4.2) par :  $\mathbf{x}_t^{j_e} = \{\mathbf{p}_t^{j_e}, \mathbf{v}_t^{j_e}, \mathbf{s}_t^{j_e}\}$ . En l'absence d'*a priori* sur la localisation des objets entrants, le point critique est de leur proposer une position pertinente  $\mathbf{p}_e^*$ .

Ces lois sont délivrées par l'algorithme 4.4, et sont nécessaires aux calculs des taux d'acceptation demandés par l'algorithme 4.3. Pour ce faire, nous calculons tout d'abord dans chaque caméra  $c$  une image de résidus dite de « faux fond », notée  $\mathbf{R}_{fb}$ , image binaire des pixels considérés comme *fond* par l'hypothèse  $\mathbf{X}$ , alors qu'ils ont été classifiés comme *forme* lors de la segmentation *fond / forme* :

$$\mathbf{R}_{fb}(\mathbf{X}, \mathbf{Z}, g, c) = \begin{cases} 1 & \text{si } \mathbf{I}_F(\mathbf{Z}, g, c) \text{ et } \overline{\mathbf{I}_M(\mathbf{X}, g, c)} \\ 0 & \text{sinon} \end{cases} \quad \forall g \in \{1, \dots, G\}, \forall c \in \{1, \dots, C\} \quad (4.34)$$

Les mouvements d'entrée sont guidés par l'image des résidus de *faux fond*, telle qu'illustré par la figure 4.5-f. Dans cette image, l'hypothèse à 3 véhicules n'explique pas le 4<sup>ième</sup> véhicule cible, ce qui fait apparaître un amas de pixels résiduels de *faux fond*, c'est à dire considérés comme *fond* par l'hypothèse, alors qu'ils ont été classifiés comme *forme* lors de la segmentation *fond / forme*. Cette observation 2D est utilisée pour guider l'exploration de l'espace d'état par le *FP MCMC*, ce qui augmente son efficacité. Guider les propositions d'entrée économise des charges de calculs, car on ne propose ainsi pas de mouvement d'entrée quand l'observation est bien expliquée par l'hypothèse, et lorsqu'on en propose un, il est initialisé dans la région de vraisemblance maximale.

#### 4.5.1 Propositions d'entrée guidées en mono-vision

Dans une installation monoculaire, un pixel est choisi au hasard dans l'image des pixels résiduels de *faux fond* :

$$\mathbf{g}_e^* \sim \frac{1}{\sum_{g=1}^G \mathbf{R}_{fb}(\mathbf{X}, \mathbf{Z}, g)} \mathbf{R}_{fb}(\mathbf{X}, \mathbf{Z}, g) \quad (4.35)$$

où  $\mathbf{g}_e$  est le vecteur contenant les coordonnées dans le plan image du point tiré. Sur la figure 4.5-f, 20 pixels ont été tirés pour illustrer la loi. Ce pixel est alors supposé être la projection dans le plan image, du centre de gravité de l'objet entrant. On en déduit sa projection au sol par homographie :

$$\mathbf{G}_e^* = H_{IG} \mathbf{g}_e^*, \quad (4.36)$$

où  $\mathbf{G}_e^*$  est le vecteur contenant les coordonnées dans le plan du sol du point tiré, respectivement notées  $x_e^*$  et  $y_e^*$ .  $H_{IG}$  est l'homographie du plan image vers le plan horizontal contenant les centres de gravité des objets de la catégorie correspondante, donc d'altitude définie par  $h_G$ , la mi-hauteur des objets. Le vecteur position de l'objet entrant est alors :

$$\mathbf{p}_e^* = [x_e^*, y_e^*, \psi_e^*], \quad (4.37)$$

avec une orientation *a priori* quelconque :  $\psi_e^* \sim \mathcal{U}(0, 2w)$ . Cette approche suppose que le pixel tiré est proche du centre de gravité de l'objet auquel on propose d'entrer, ce qui peut conduire à de grandes incertitudes sur la distance de l'objet à la caméra, notamment lorsque celle-ci est positionnée à faible altitude. De plus, dans une approche multi-catégorie, l'altitude du plan dépend évidemment de la catégorie de l'objet candidat. Ces inconvénients ne concernent néanmoins que l'initialisation de l'objet, et il faudra bien sûr laisser à l'algorithme le temps de converger pour valider ou invalider ce nouvel objet candidat.

### 4.5.2 Propositions d'entrée guidées en multi-vision

Le processus d'entrée précédemment décrit dans le cas d'une installation monoculaire, souffre d'une grande incertitude sur la distance de l'objet à la caméra, causée par l'effet projectif, notamment lorsque celle-ci est positionnée à faible altitude. Dans une installation multi-oculaire, cette incertitude peut être diminuée par triangulation, ce que nous réalisons par le processus suivant :

1. Calculer la loi  $\mathbf{K}_{fb}(\mathbf{X}, \mathbf{Z}, g, c)$ , somme par colonne de  $\mathbf{R}_{fb}(\mathbf{X}, \mathbf{Z}, g, c)$ , image des pixels résiduels de *faux fond*  $\forall c \in \{1, \dots, C\}$ .
2. Tirer une colonne dans chaque caméra :

$$\mathbf{K}_c^* \sim \mathbf{K}_{fb}(\mathbf{X}, \mathbf{Z}, g, c), \forall c \in \{1, \dots, C\} \quad (4.38)$$

Soit  $\delta_c^*$  cette droite,  $\forall c \in \{1, \dots, C\}$ .

3. Cette droite est alors supposée être la projection dans le plan image, d'une droite contenue dans le plan horizontal des centres de gravité :

$$\Delta_c^* = H_{IG,c} \delta_c^*, \forall c \in \{1, \dots, C\}, \quad (4.39)$$

où  $H_{IG,c}$  est l'homographie du plan image vers le plan horizontal contenant les centres de gravité des objets de la catégorie correspondante, donc d'altitude définie par  $h_G$ , la mi-hauteur des objets.

4. Proposer un objet entrant au point appartenant à ce plan, minimisant sa distance quadratique aux droites  $\Delta_c^*$ .

Le processus est illustré par la figure 4.8. L'hypothèse courante (contours jaunes sur les images du haut) considère seulement 3 personnes, c'est à dire que la personne en veste bleue n'est pas expliquée. Il en résulte des amas importants de pixels sur les images de *faux fond* (ligne du centre). Les lignes verticales sont les  $\delta_c^*$ , colonnes tirées dans chaque image de résidus, projections dans les plans images des droites  $\Delta_c^*$ . 20 échantillons ont été tirés pour illustrer la densité. Les astérisques rouges illustrent l'intersection des droites obtenues par homographies de ces lignes dans le plan horizontal des centres de gravité. Ce processus permet au nombre de caméras en service d'augmenter la précision du processus d'entrée d'un nouvel objet.

## 4.6 Guidage des mouvements de sortie

### 4.6.1 Mouvements de sortie guidés par l'observation courante

A chaque itération  $n$  de l'instant  $t$ , on calcule pour chaque objet  $j$  dans chaque caméra  $c$ , le taux de *fausse forme* noté  $f_{t,c}^{j,n}$ , c'est à dire le nombre de pixels couverts par l'enveloppe convexe de la projection dans la caméra  $c$  de l'objet  $j$ , mais classifiés comme *fond* par la segmentation *fond / forme*, divisé par la surface dudit objet, selon l'équation (4.40) :

$$f_{t,c}^{j,n} = \frac{1}{|\mathcal{R}_{t,c}^{j,n}|} \sum_{g \in \mathcal{R}_{t,c}^{j,n}} \overline{\mathbf{I}_F(\mathbf{Z}, g, c)}, \quad (4.40)$$

où  $\mathcal{R}_{t,c}^{j,n}$  est la région de l'image couverte par l'enveloppe convexe de la projection de l'objet  $j$ ,  $\forall j \in \{1, \dots, J_t^n\}$ , dans l'image de la caméra  $c$ ,  $\forall c \in \{1, \dots, C\}$ . Considérant que la plus-value du multi-vue est une observabilité



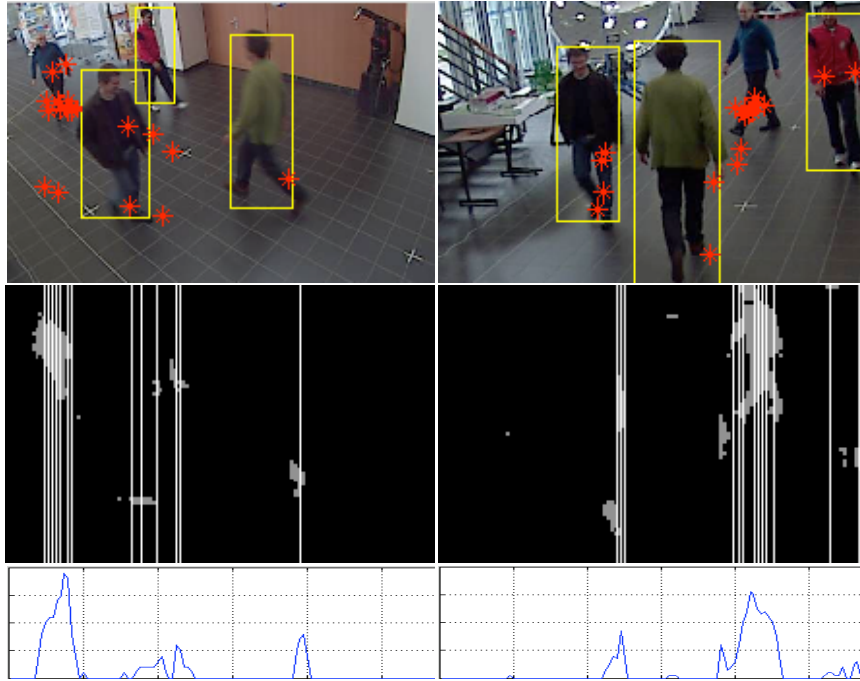


FIGURE 4.8 – En haut : hypothèse courante comportant 3 personnes. La personne en veste bleue n'est pas expliquée par cette hypothèse. Astérisques rouges : localisations au sol d'un 4<sup>ième</sup> objet proposé par le processus d'entrée. Au centre : images de *faux fond*  $\mathbf{R}_{fb}(\mathbf{X}, \mathbf{Z}, g, c)$ , c'est à dire pixels couverts par la projection d'aucun objet de l'hypothèse, mais classifiés comme *forme* lors de la segmentation *fond / forme*. Les colonnes correspondant à un échantillonnage de ces pixels est tracé (lignes blanches verticales  $\delta_c^*$ ). En bas :  $\mathbf{K}_{fb}(\mathbf{X}, \mathbf{Z}, g, c)$ , densité de ces pixels par colonne dans l'image, mettant en évidence les amas importants de pixels concentrés sur la personne en veste bleue.

supérieure à celle par du mono-vue, on ne retient que le meilleur de ces taux de *fausse forme*, on ne retient que le meilleur sur toutes les caméras :

$$f_t^{j,n} = \max_c f_{t,c}^{j,n}, \quad (4.41)$$

On pourrait alors conditionner la loi de proposition des sorties à ce taux de *fausse forme*  $f_t^{j,n}$ , de façon à proposer d'autant plus de sorties, que  $f_t^{j,n}$  est faible. Cette approche présente deux inconvénients :

- ▷ Contributions multiples d'un même pixel à plusieurs objets : en cas d'occultation, c'est à dire lorsque les deux régions  $\mathcal{R}_{t,c}^{j,n}$  et  $\mathcal{R}_{t,c}^{k,n}$  de deux objets  $j$  et  $k$  ont une intersection non nulle dans la caméra  $c$ , les pixels de l'intersection vont contribuer aux deux objets. Ceci n'est pas souhaitable dans le cas d'objets opaques. Nous proposons en section 4.6.2 une solution à ce problème.
- ▷ Pas de cohérence temporelle : dès qu'un objet cible est mal observé (mal segmenté du fond, ou occulté par un autre objet), on lui propose souvent de sortir, et cette sortie est facilement acceptée. Dès que l'objet cible est à nouveau observable le *FP RJ-MCMC* propose l'entrée d'un nouvel objet candidat, mais avec une nouvelle identité. En absence d'un mécanisme de reconnaissance, la cohérence temporelle du suivi de l'objet cible est alors perdue. Pour éviter ce problème, nous guidons les sorties de chaque objet à partir de sa *vitalité* définie en section 4.6.3.

#### 4.6.2 Contribution unique des pixels

Les objets que nous suivons étant opaques, le coefficient  $f_{t,c}^{j,n}$  ne doit être calculé que sur la base des pixels visibles de cet objet, c'est à dire non occultés par un autre objet  $k$ , placé entre l'objet  $j$  et la caméra  $c$ . Pour ce faire, nous redéfinissons  $\mathcal{R}_{t,c}^{j,n}$  de la façon suivante :

$$\mathcal{R}_{t,c}^{j,n} = \mathcal{R}_{t,c}^{j,n} \setminus \bigcup_{k \in \mathcal{K}_{t,c}^{j,n}} \mathcal{R}_{t,c}^{k,n}, \forall j \in \{1, \dots, J_t^n\}, \forall c \in \{1, \dots, C\} \quad (4.42)$$

où  $\mathcal{K}_{t,c}^{j,n}$  est l'ensemble des objets plus proches de la caméra  $c$  que l'objet  $j$  ne l'est, donc susceptibles d'occulter l'objet  $j$  dans la caméra  $c$ . Ce calcul requiert donc que les objets de chaque particule soient ordonnés selon leur distance à chaque caméra  $c$ . Pour ce faire, la position absolue de chaque caméra  $c$  doit être connue.

#### 4.6.3 Vitalité des objets présents.

Comme argumenté dans l'introduction de ce chapitre, nous proposons ici d'affecter à chaque objet une variable additionnelle, la *vitalité*, mise à jour sur la base des observations. Elle permet de rendre réglable le nombre d'observations nulles auquel un objet survivra et conservera son identité. Elle a pour but d'assurer la cohérence temporelle du nombre d'objets présents dans le champ de vision et de leurs identités respectives, en compilant l'historique de leurs observabilités respectives. Le domaine de variation de la *vitalité* s'étend de 0 à 1. Notons  $j$  un objet candidat générique,  $j \in \{1, \dots, J_t^n\}$ . A chaque itération  $n$  de l'instant  $t$ , on calcule pour chaque objet  $j$  le taux de *fausse forme*  $f_t^{j,n}$ , conformément à l'équation (4.40). L'incrément de *vitalité* de l'objet  $j$ , noté  $\lambda_t^j$  est calculé sur toute la chaîne de particules (après *Burn-in*, c'est à dire élimination des  $N_B$  premières particules), comme une fonction sigmoïde de  $f_t^{j,n}$  :

$$\lambda_t^j = k_d \cdot \sum_{n=1}^N \frac{e^{-k_r \cdot (f_t^{j,n} - r_f)} - 1}{e^{-k_r \cdot (f_t^{j,n} - r_f)} + 1}, \forall j \in \{1, \dots, J_t^n\}, \quad (4.43)$$

où  $r_f$  est le paramètre d'inflexion de la courbe des taux de *fausse forme*, c'est-à-dire la valeur de  $f_t^{j,n}$  donnant un incrément nul, et  $k_r$  est le paramètre de pente de la courbe. L'équation (4.43) produit un incrément positif si  $f_t^{j,n} < r_f$ , sinon négatif, ce qui permet à la *vitalité* de compiler l'historique des vraisemblances de l'objet  $j$  au cours du temps. Le coefficient  $k_d$ , permettant d'ajuster la dynamique de la *vitalité*, est calculé selon l'équation (4.44) :

$$k_d = \frac{1 - \Lambda_0}{n_s \cdot C \cdot N}, \quad (4.44)$$

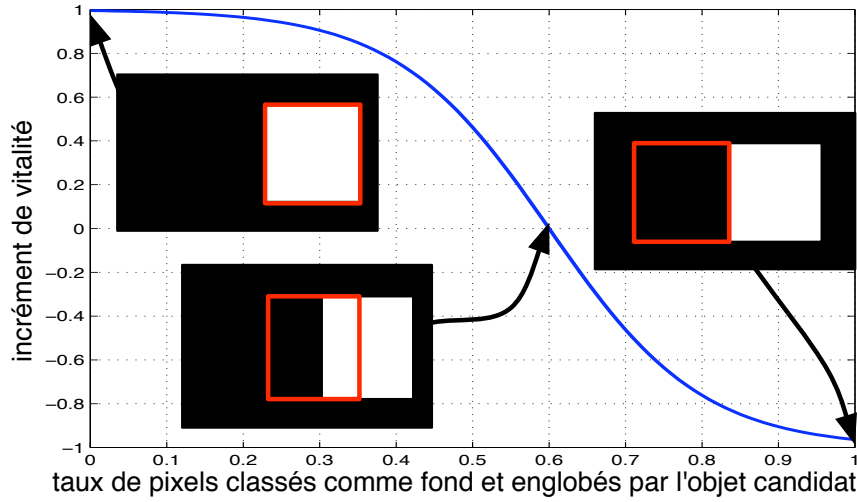


FIGURE 4.9 – Incrément de *vitalité*  $\lambda_t^j$  en fonction de  $f_t^{j,n}$  taux de *fausse forme* de l'objet  $j$  en monovision, avec  $k_d = 1$ ,  $r_f = 0.6$  et  $k_r = 10$ .

où  $\Lambda_0$  est la *vitalité* initiale d'un objet entrant,  $n_s$  indique le nombre d'images d'invisibilité totale auquel un objet avec le maximum de *vitalité* peut survivre. Une telle invisibilité est généralement causée par l'occultation totale de l'objet par l'arrière-plan : par exemple lorsqu'un véhicule passe derrière un pont ou un arbre. Ce paramètre  $n_s$  permet à l'utilisateur d'ajuster la dynamique de la *vitalité* en fonction de la durée des occultations possibles. La *vitalité* de chaque objet est finalement mise à jour pour le prochain instant  $t + 1$ , par l'équation :

$$\Lambda_{t+1}^j = \begin{cases} \min(\Lambda_t^j + \lambda_t^j, 1) & \text{si } z_j \\ \max(\Lambda_t^j + \lambda_{out}, 0) & \text{sinon} \end{cases}, \quad (4.45)$$

où  $z_j$  est une variable binaire fixée à 1 si le centre de gravité de l'objet  $j$  est situé dans la zone de suivi, sinon à 0. Si l'objet  $j$  est en dehors de la zone de suivi, sa *vitalité* est incrémentée de  $\lambda_{out}$ . Les valeurs choisies pour les expériences, indiquées dans le tableau 4.2, conduisent aux incréments de *vitalité* illustrés sur la figure 4.9.

#### 4.6.4 Propositions de sortie d'un objet guidées par sa *vitalité*.

A chaque instant  $t$ , la probabilité de sortie de l'objet  $j$  à l'instant  $t$ , notée  $Q(j|s, \Lambda_t^j)$  est déterminée par  $\Lambda_t^j$ , sa *vitalité* à l'instant  $t$ , selon l'équation (4.46) :

$$Q(j|s, \Lambda_t^j) \propto \left(1 + e^{k_v \cdot (\Lambda_t^j - \Lambda_0)}\right)^{-1}. \quad (4.46)$$

Le paramètre d'inflexion de la sigmoïde est choisi égal à  $\Lambda_0$ , ce qui assure la réversibilité des mouvements d'entrée et de sortie, donc la compatibilité à l'algorithme de Metropolis-Hastings. De moins en moins de propositions de sortie seront générées à mesure que la *vitalité* de l'objet  $j$  croît, ce qui lui évitera de quitter la scène immédiatement lorsque son observabilité est faible à l'instant courant  $t$ . Au contraire, la *vitalité* lui permet de survivre plusieurs images d'occultation totale ou de mauvaise segmentation *fond-forme*.  $k_v$  est le paramètre de pente de la sigmoïde. Le même mécanisme est utilisé pour le soleil, avec une dynamique plus lente déterminée par un  $n_s$  plus élevé (voir tableau 4.2).

## 4.7 Mouvements guidés : mise en œuvre

- ▷ On a argumenté la nécessité de guider les mouvements, notamment d'entrée, en section 4.3.4.

TABLE 4.2 – Paramètres de calcul de la *vitalité* des objets.  $n_s = 25$  images permet à un objet de survivre à une occultation totale s'étendant sur 25 images, soit une seconde à 25 images/s.

	$\Lambda_0$	$\lambda_{out}$	$r_f$	$n_s$	$k_r$	$k_v$
<b>objet</b>	0.2	-0.1	0.6	10	10	10
<b>soleil</b>	0.2	.	0.4	50	10	10

- ▷ On a proposé en section 4.5 des méthodes de guidage des propositions d'entrées, les rendant dépendantes de l'observation courante  $\mathbf{Z}_t$ , et de la particule précédente  $\mathbf{X}_t^{n-1}$
- ▷ On a introduit en section 4.6 la notion de *vitalité* définie pour chaque objet  $j$ , et mise à jour à chaque pas temporel à partir de l'observation courante  $\mathbf{Z}_t$ , et de toutes les instances  $\mathbf{x}_t^{j,n-1}$  de l'objet  $j$  acceptées dans la chaîne. On a alors défini un mécanisme de guidage des propositions de sortie de chaque objet dépendant de sa propre *vitalité*.
- ▷ L'approximation de la loi cible par l'échantillonneur de Metropolis-Hastings que nous utilisons, converge vers la cible quelle que soit la loi de proposition des échantillons, pourvu que les formules de calcul du taux d'acceptation (4.15), (4.17), et (4.19), soient respectées.

On peut alors alimenter l'algorithme 4.3 par les lois de proposition guidées, délivrées par l'algorithme 4.4. Les propositions d'entrées sont guidées à partir de l'observation courante  $\mathbf{Z}_t$ , et de la particule précédente  $\mathbf{X}_t^{n-1}$ . Les propositions de sorties de chaque objet  $j$  sont alors guidées par la *vitalité*  $\Lambda_t^j$ , selon l'équation 4.46.

---

**Algorithme 4.4** Gestion des mouvements du *FP RJ-MCMC* guidée par l'observation

---

**Entrées :**

- fréquence de proposition d'une entrée  $f_e$ , relative aux mises à jour.
- fréquence de proposition d'une sortie  $f_s$ , relative aux mises à jour.
- état  $\mathbf{X}_t^{n-1}$  à l'itération précédente de la chaîne courante
- observation  $\mathbf{Z}_t$  à l'instant courant  $t$ .
- *vitalité*  $\Lambda_t^j$  de chaque objet  $j$  vital :  $j \in \{1, \dots, J_t\}$
- ensemble des types de mouvements  $\mathcal{M} = \{e, u, s\}$  où  $e$  est un mouvement d'entrée,  $u$  représente la mise à jour de l'état d'un des objets de  $\mathbf{X}_t^{n-1}$ ,  $s$  représente la sortie d'un des objets de  $\mathbf{X}_t^{n-1}$ .

**Calcul de la loi de mouvements non normalisée**  $q(m|\mathbf{X}_t^{n-1}, \mathbf{Z}_t)$  :

- entrées :  $Q(e|\mathbf{X}_t^{n-1}, \mathbf{Z}_t)$  selon méthode décrite en section 4.5.
- mises à jour :  $Q(u|\mathbf{X}_t^{n-1}, \mathbf{Z}_t) = J_t^{n-1}$
- sorties :  $Q(s|\{\Lambda_t^j\}_{j=1}^{J_t})$  selon méthode décrite en section 4.6.

**Calcul de la loi de choix de l'objet affecté**  $q(j|\mathbf{X}_t^{n-1}, \mathbf{Z}_t, m)$  :

- choix objet mis à jour :  $Q(j|\mathbf{X}_t^{n-1}, \mathbf{Z}_t, u) = 1/J_t^{n-1}, \forall j \in \{1, \dots, J_t^{n-1}\}$
- choix objet sorti :  $Q(j|\Lambda_t^j, s)$  selon méthode décrite en section 4.6.

**Normalisation :**

- normaliser :  $Q(m|\mathbf{X}_t^{n-1}, \mathbf{Z}_t) = Q(m|\mathbf{X}_t^{n-1}, \mathbf{Z}_t) (f_e + J_t^{n-1}(1 + f_s))^{-1}$
- calculer  $\Sigma = Q(e|\mathbf{X}_t^{n-1}, \mathbf{Z}_t) + \sum_{m \neq e} \sum_{j=1}^{J_t^{n-1}} Q(m|\mathbf{X}_t^{n-1}, \mathbf{Z}_t) Q(j|\mathbf{X}_t^{n-1}, \mathbf{Z}_t, m)$
- normaliser :  $Q(j|\mathbf{X}_t^{n-1}, \mathbf{Z}_t, m) = Q(j|\mathbf{X}_t^{n-1}, \mathbf{Z}_t, m) \Sigma^{-1}$

**Sorties :**

- $q(m) = q(m|\mathbf{X}_t^{n-1}, \mathbf{Z}_t), m \in \mathcal{M} = \{e, u, s\}$
  - $q(j|m) = q(j|\mathbf{X}_t^{n-1}, \mathbf{Z}_t, \Lambda_t^j, m), m \in \mathcal{M} = \{e, u, s\}, j \in \{1, \dots, J_t^{n-1}\}$
  - $q(\mathbf{x}^{j_e})$ , loi de probabilité de l'état *a priori* d'un objet entrant.
-

## 4.8 Potentiel d'interaction

Comme la fonction de vraisemblance d'avant plan permet à un objet totalement occulté de survivre, nous devons l'empêcher de rester « caché » derrière un autre objet.

### 4.8.1 Modélisation des interactions dans la loi dynamique

En l'absence d'interaction, chaque objet se déplace selon sa propre dynamique, indépendamment des autres objets. La loi dynamique du système joint peut donc s'écrire selon l'équation (4.47) :

$$p(\mathbf{X}_t | \mathbf{X}_{t-1}) = \prod_{j=1}^J p(\mathbf{x}_t^j | \mathbf{x}_{t-1}^j), \quad (4.47)$$

où  $J$  est le nombre d'objets présents. Pour prendre en compte le fait que les objets se déplacent habituellement à distance les uns des autres, Khan *et al.* ont proposé d'injecter dans cette loi de dynamique un potentiel modélisant les interactions entre objets (KBD05).

$$p(\mathbf{X}_t | \mathbf{X}_{t-1}) = \prod_{j=1}^J p(\mathbf{x}_t^j | \mathbf{x}_{t-1}^j) \prod_{i,j \in \mathcal{E}} \psi(\mathbf{x}_t^i, \mathbf{x}_t^j), \quad (4.48)$$

où  $\psi(\mathbf{x}_t^i, \mathbf{x}_t^j)$  est un potentiel d'interaction entre les objets  $i$  et  $j$ , et  $\mathcal{E}$  est l'ensemble des couples d'objets susceptibles d'interagir. Par souci de simplicité, ce potentiel est limité aux interactions entre deux objets, mais la même méthode peut être généralisée à un nombre quelconque d'objets.

Pour suivre des personnes, (Y008) propose un potentiel anisotrope, basé sur la distance de Mahalanobis entre objets. Nous adoptons cette distance anisotrope, indispensable en suivi de véhicule, car leur longueur est beaucoup plus importantes que leur largeur (environ 8 fois pour les poids lourds). La distance entre les centres de gravité de deux véhicules roulant côte à côte sur deux voies voisines est d'environ 3 mètres, tandis que la même distance entre deux véhicules roulant l'un derrière l'autre sur la même voie est matériellement impossible. En outre, l'interaction entre les deux véhicules dépendra de leurs dimensions. Pour être compatible avec ces deux conditions, le calcul du potentiel d'interaction  $w_{inter}^*$  est basé sur la distance anisotrope entre chaque paire de véhicules, définie comme suit : chaque véhicule est approximé par une distribution gaussienne bidimensionnelle, dont la matrice de covariance est constituée des moments de masse d'ordre deux. La distance de Mahalanobis entre ces deux distributions gaussiennes bidimensionnelles est ensuite calculée :

$$d_{ij} = (\Delta_{ij}^T (\mathbf{C}_i \cdot \mathbf{C}_j)^{-1} \Delta_{ij})^{1/2} \quad (4.49)$$

où  $\Delta_{ij} = |\mathbf{x}_t^i - \mathbf{x}_t^j|$  est le vecteur de différence de position des objets  $i$  et  $j$ ,  $\mathbf{C}_i$  et  $\mathbf{C}_j$  leurs matrices de covariance respectives. On traite cette distance par la fonction sigmoïde (4.50), générant des valeurs tendant vers 1 pour les objets lointains et vers 0 pour les distances inter-objets ne respectant pas les contraintes matérielles.

$$\psi(\mathbf{x}_t^i, \mathbf{x}_t^j) = \frac{1}{1 + e^{-k_s \cdot (d_{ij} - d_s)}}, \quad (4.50)$$

où  $d_s$  est la distance « seuil » et  $k_s$  un coefficient permettant d'ajuster la pente de la sigmoïde autour de  $d_s$ . Nous choisissons  $d_s = 2$  et  $k_s = 20$  dans nos expérimentations, comme représenté sur la figure 4.10.

### 4.8.2 Poids d'interaction

Si on remplace chaque composante de l'équation (2.8), par l'expression établie en (4.48), on obtient :

$$p(\mathbf{X}_t | \mathbf{Z}_{1:t-1}) \approx \prod_{i,j \in \mathcal{E}} \psi(\mathbf{x}_t^i, \mathbf{x}_t^j) \frac{1}{N} \sum_{n=1}^N \prod_{j=1}^J p(\mathbf{x}_t^j | \mathbf{x}_{t-1}^{j,n}). \quad (4.51)$$

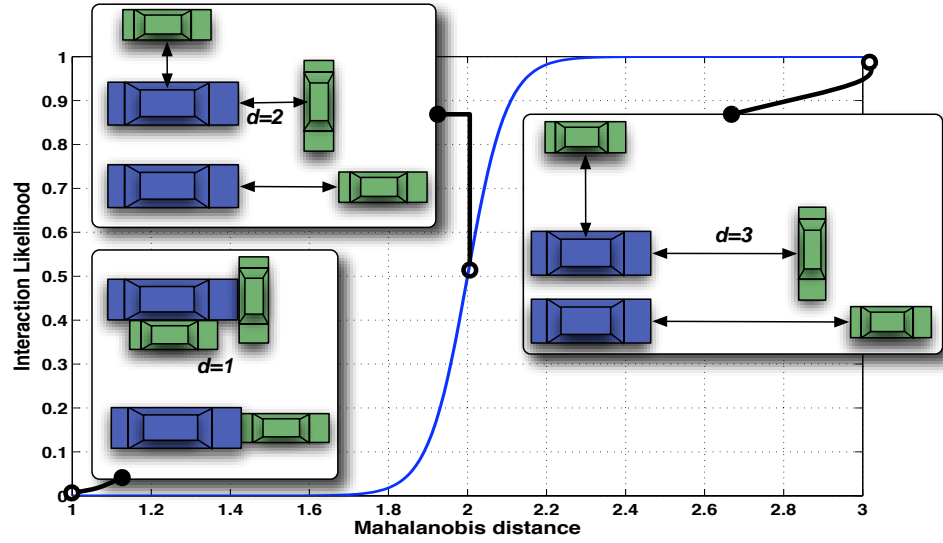


FIGURE 4.10 – Fonction sigmoïde définissant la probabilité d’interaction à partir de la distance anisotrope inter-véhicule  $d$ . Quelques configurations de véhicules correspondant à trois valeurs de  $d$ , sont dessinées dans le cas de deux véhicules. Le véhicule bleu est nettement plus grand que le vert pour montrer l’indépendance de la distance à la taille du véhicule.  $d$  vaut 1 si les deux véhicules sont en contact, quelles que soient leurs faces en contact et leurs dimensions respectives. La sigmoïde est tracée pour les valeurs choisies  $k = 20$  et  $d_s = 2$ , qui correspond à l’image en haut à gauche, permettant à deux véhicules de rouler côte à côte sur deux voies adjacentes, tout en pénalisant les configurations trop proches (image en bas à gauche).

Le potentiel d’interaction a été sorti de la mixture car il est indépendant de l’état antérieur  $\mathbf{X}$ . Il peut ainsi être traité comme une pondération intervenant au même niveau que la vraisemblance. Nous le notons :

$$w_{inter}(\mathbf{X}_t) = \prod_{i,j \in \mathcal{E}} \psi(\mathbf{x}_t^i, \mathbf{x}_t^j) \quad (4.52)$$

La prise en compte des interactions se fait alors en remplaçant les équations (4.11) et (4.12) par :

$$w^* = P(\mathbf{Z}_t | \mathbf{X}_t^*) w_{inter}(\mathbf{X}_t^*) \quad (4.53)$$

et

$$w_t^{n-1} = P(\mathbf{Z}_t | \mathbf{X}_t^{n-1}) w_{inter}(\mathbf{X}_t^{n-1}), \quad (4.54)$$

qui intègrent la vraisemblance de l’observation et le potentiel d’interaction. Ces poids sont alors utilisés pour calculer le taux d’acceptation, conformément aux équations (4.15), (4.17), et (4.19).

## 4.9 Extension de la méthode au suivi de véhicules

### 4.9.1 Extension du vecteur d’état

Du fait qu’un véhicule réel se commande en accélération longitudinale (accélérateur / frein), et en accélération transversale (par la combinaison de la vitesse et de l’angle de braquage des roues directrices), il est intéressant d’étendre le vecteur d’état de chaque objet :

$$\mathbf{x}_t^{j,n} = \{\mathbf{p}_t^{j,n}, \mathbf{v}_t^{j,n}, \mathbf{a}_t^{j,n}, \mathbf{s}_t^{j,n}\} \quad (4.55)$$

Les vecteurs  $\mathbf{p}_t^{j,n}$ ,  $\mathbf{v}_t^{j,n}$ , et  $\mathbf{s}_t^{j,n}$  sont définis comme dans (4.2), tandis que la commande du conducteur au temps  $t$  est définie par  $\mathbf{a}_t^{j,n} = \{\alpha_t^{j,n}, \gamma_t^{j,n}\}$ , où  $\alpha_t^{j,n}$ , est l’angle de braquage et  $\gamma_t^{j,n}$ , l’accélération longitudinale.

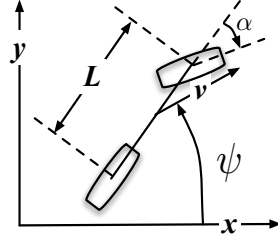


FIGURE 4.11 – Modélisation d'un véhicule par bicycle cinématique, valide dans le cas des courbes de grand rayon *i.e.*  $\text{rayon} \gg L$ , où  $L$  est l'empattement du véhicule. Dans ce cas, l'angle de lacet (orientation du véhicule autour de l'axe normal au plan du sol) du véhicule noté  $\psi$  peut être confondu avec l'orientation de la vitesse du centre de gravité du véhicule. Les paramètres de position du centre de gravité du véhicule sont notés  $x$  et  $y$ . L'amplitude de la vitesse du centre de gravité du véhicule est notée  $v$ , l'angle de braquage de l'essieu directeur est noté  $\alpha$ .

#### 4.9.2 Modèle dynamique des véhicules

Afin d'améliorer l'efficacité du suivi, nous limitons l'espace d'état à l'étendue que peuvent physiquement atteindre les objets suivis. A cette fin, chaque objet est contrôlé par le modèle dynamique correspondant à sa catégorie. Si l'objet est supposé être un piéton, sa dynamique est décrite par les équations (4.3) et (4.4). Si l'objet est supposé être un véhicule, sa dynamique obéit également aux équations (4.3) et (4.4), et à deux équations supplémentaires modélisant le roulement sans glissement d'un véhicule à roues avant directrices, connu sous le nom de modèle bicycle (Gil92; GCMT06), illustré par la figure 4.11. Les équations (4.56) et (4.57) modélisent respectivement la dynamique longitudinale et transversale du véhicule d'empattement  $L$  :

$$v_t^{j,n} = v_{t-1}^{j,n} + T \cdot \gamma_{t-1}^{j,n}, \quad (4.56)$$

$$\psi_t^{j,n} = \psi_{t-1}^{j,n} + T \cdot v_{t-1}^{j,n} \cdot \tan(\alpha_{t-1}^{j,n}) / L, \quad (4.57)$$

#### 4.9.3 Loi de proposition de commande d'un véhicule

Dans le cas où l'objet  $j$  est un véhicule, on tire une proposition de commande du conducteur, selon (4.58) :

$$p(\mathbf{a}^* | \mathbf{a}_{t-1}^{j,r}) = \mathcal{N}(0, \text{diag}([\sigma_l^2, \sigma_a^2])), \quad (4.58)$$

où  $\sigma_l$  est l'écart-type des accélérations longitudinales pratiquées par les conducteurs,  $\sigma_a$  est l'écart-type des angles de braquage pratiqués par les conducteurs. Les équations du bicycle (4.56) et (4.57) sont alors appliquées à l'objet  $j$  pour mettre à jour sa vitesse et son angle de lacet. Les commandes pratiquées par les conducteurs dépendent de la catégorie de véhicule proposée, et sont données tableau 4.3. On applique enfin les équations de la dynamique (4.3) et (4.4) pour mettre à jour la position  $\mathbf{p}_t^{j,n}$  de l'objet  $j$ . La forme de l'objet reste mise à jour selon (4.6).

### 4.10 Catégorisation des objets intégrée au suivi

Certaines applications requièrent le suivi simultané d'objets pouvant appartenir à plusieurs catégories, d'apparences différentes. C'est le cas du suivi de véhicules dans le trafic routier. Compte tenu de la fonction de vraisemblance utilisée, nous avons expérimenté deux approches :

- ▷ Mener séquentiellement le suivi puis la catégorisation des objets en deux modules indépendants.
- ▷ Mener simultanément le suivi et la catégorisation des objets, en intégrant cette dernière au filtre.



TABLE 4.3 – Exemples de réglages de l’empattement des véhicules et des écarts-types des commandes conducteur, en fonction de la catégorie des objets.

catégorie véhicule	empattement L (m)	écart type accélération $\sigma_l (m.s^{-2})$	écart type braquage $\sigma_a (deg)$
moto	1.2	4	2
voiture	2.5	4	2
camionnette	4.5	2	2
poids lourd	10	1	1

Du fait de l’hétérogénéité des dimensions des véhicules, la première option impose d’appliquer une forte dynamique aux dimensions des objets. Rien n’empêche alors d’arriver à la situation illustrée par la figure 4.12, où deux objets candidats expliquent conjointement une seule cible. Rien n’empêche non plus un unique objet candidat d’expliquer deux objets cibles distincts mais dont les projections dans l’image sont connexes. Pour cette raison, nous choisissons de mener simultanément le suivi et la catégorisation des objets, en confiant ces deux fonctions au *FP MCMC*. L’intérêt de ce choix est double. Il permet d’une part de maintenir et propager des hypothèses appartenant à différentes catégories, celle-ci étant traitée comme une variable aléatoire (discrète) au même titre que les autres attributs (continus) de l’objet. Il permet d’autre part de générer des prédictions dynamiques spécifiques à chaque catégorie d’objet, permettant à la dynamique temporelle des objets de contribuer à leur catégorisation au même titre que leur géométrie. Cela est d’autant plus bénéfique que les objets ont des dynamiques différentes, telles que poids lourd/voiture sur route sinueuse, et surtout dans le cas où le modèle géométrique n’est pas suffisamment discriminant (piéton/cycliste).

#### 4.10.1 Intégration de la catégorie dans le vecteur d’état

Afin de réaliser le suivi et la catégorisation simultanés des objets, le vecteur d’état de chaque objet défini précédemment par (4.2), est enrichi de sa catégorie. La définition du vecteur d’état (4.2) est donc remplacée par :

$$\mathbf{x}_t^{j,n} = \{c_t^{j,n}, \mathbf{p}_t^{j,n}, \mathbf{v}_t^{j,n}, \mathbf{a}_t^{j,n}, \mathbf{s}_t^{j,n}\} \quad (4.59)$$

La catégorie de l’objet  $j$  à l’itération  $n$  est définie par  $c_t^{j,n}$ , une variable aléatoire discrète pouvant prendre un des états contenus dans l’ensemble de catégories  $C=\{\text{piéton}, \text{moto}, \text{voiture}, \text{camionnette}, \text{poids lourd}\}$  par exemple.

#### 4.10.2 Mise à jour conjointe objet et catégorie

Si on note  $\mathbf{y} = \{\mathbf{p}, \mathbf{v}, \mathbf{a}, \mathbf{s}\}$ , on peut écrire  $\mathbf{x} = \{c, \mathbf{y}\}$ . La loi dynamique de l’objet peut se développer :

$$p(\mathbf{x}_t^j | \mathbf{x}_{t-1}^j) = p(c_t^j, \mathbf{y}_t^j | c_{t-1}^j, \mathbf{y}_{t-1}^j) = p(c_t^j | c_{t-1}^j, \mathbf{y}_{t-1}^j) p(\mathbf{y}_t^j | c_t^j, c_{t-1}^j, \mathbf{y}_{t-1}^j), \quad (4.60)$$

qui peut se simplifier en :

$$p(\mathbf{x}_t^j | \mathbf{y}_{t-1}^j) = p(c_t^j | c_{t-1}^j) p(\mathbf{y}_t^j | c_t^j, \mathbf{x}_{t-1}^j), \quad (4.61)$$

où  $p(c_t^j | c_{t-1}^j)$  est la loi de probabilité de changement de catégorie d’un objet, tandis que  $p(\mathbf{y}_t^j | c_t^j, \mathbf{x}_{t-1}^j)$  est la dynamique de l’objet  $j$ , conditionnée à sa catégorie. Ceci permet de définir des modèles dynamiques distincts, spécifiques à chaque catégorie. La loi de probabilité de changement de catégorie d’un objet est définie par :

$$P(c^* = i | c_{t-1}^{j,r} = j) = \Theta_{ij}, \forall i, j \in \{1, \dots, |C|\} \quad (4.62)$$

où  $\Theta_{ij}$  est la probabilité de transition de la catégorie  $i$  à la catégorie  $j$ . Les catégories sont classées par ordre croissant de taille. Par exemple si  $C=\{\text{moto}, \text{voiture}, \text{camionnette}, \text{poids lourd}\}$ ,  $c = 1$  si pour une moto, etc... La matrice de probabilités de transitions  $\Theta$  n’autorise que le passage vers les catégories les plus proches en





FIGURE 4.12 – Suivi sans catégorisation. Une forte dynamique de dimensions des objets est nécessaire. Il en résulte plusieurs dysfonctionnements. Explication de 2 cibles par un candidat unique : un seul objet #5 suit 2 cibles proches à l'image #342 ; de même un seul objet #13 suit 2 cibles s'occultant à l'image #680. Explication d'une cible unique par 2 candidats : le poids-lourd image #450 est expliqué par les objets candidats #7 et #12. Enfin, forte sensibilité au bruit d'observation, de l'estimation de forme et de position d'un objet : une erreur de près de 20 mètres est commise sur la l'estimation de position de l'objet #12 image 461 !

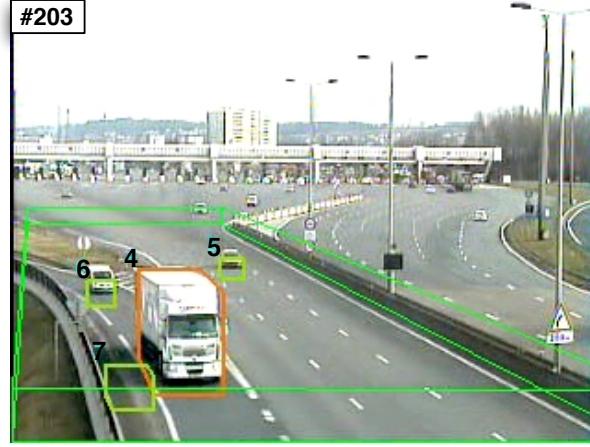


FIGURE 4.13 – Instanciation erronée d'un objet candidat (7) sur l'ombre du poids-lourd (4).

termes de dimensions. Avec l'exemple précédent où  $|\mathcal{C}| = 4$ ,  $\Theta$  est définie par (4.63), où  $t_c$  est la probabilité de transition vers une des catégories les plus proches en termes de dimensions.

$$\Theta = \begin{pmatrix} 1 - 2.t_c & 2.t_c & 0 & 0 \\ t_c & 1 - 2.t_c & t_c & 0 \\ 0 & t_c & 1 - 2.t_c & t_c \\ 0 & 0 & 2.t_c & 1 - 2.t_c \end{pmatrix} \quad (4.63)$$

Le mouvement de mise à jour d'un objet défini dans la section 4.3.1, est alors modifié comme décrit ci-dessous :

- ▷ On tire au hasard un objet  $j$  de la particule  $\mathbf{X}_t^{n-1}$  à mettre à jour. On tire au hasard dans la chaîne précédente  $\{\mathbf{X}_{t-1}^n\}_{n=1}^N$  une particule  $\mathbf{X}_{t-1}^r$ ,  $r \in \{1, \dots, N\}$ , qui contient une instance de l'objet  $j$  à  $t - 1$  notée  $\mathbf{x}_{t-1}^{j,r}$ .
- ▷ On prédit sa catégorie  $c^*$  selon la loi  $p(c_t^j | c_{t-1}^j)$
- ▷ Si  $c^*$  propose que l'objet  $j$  soit un véhicule, on tire une proposition de commande du conducteur, selon (4.58). Les équations du bicycle (4.56) et (4.57) sont alors appliquées à l'objet  $j$  pour mettre à jour sa vitesse et son angle de lacet. Si  $c^*$  propose que l'objet  $j$  soit un piéton, on tire une proposition de vitesse selon (4.5).
- ▷ On applique les équations de la dynamique (4.3) et (4.4) pour mettre à jour la position  $\mathbf{p}_t^{j,n}$  de l'objet  $j$ . La forme de l'objet est mise à jour selon (4.6).

## 4.11 Modélisation et suivi de l'illumination

L'ombre des objets portée sur les éléments du *fond* interfère avec la segmentation et la description des objets. Ceci diminue la précision de suivi, car les estimations de position des objets sont attirées vers son ombre. En outre, ce phénomène provoque des faux positifs dans le suivi, lorsque le traqueur instancie un objet inexistant sur une ombre, comme illustré par la figure 4.13. Pour éviter ces dysfonctionnements, l'ombre doit être prise en compte. Nous avons envisagé deux approches :

- ▷ Segmentation binaire de l'image en 2 classes : *fond* et *forme*, telle qu'illustrée par la figure 4.14 colonne du milieu. On voit que les ombres couvrent une surface aussi importante que celle des objets. Dans cette option, il est donc indispensable de modéliser l'éclairement et l'ombre portée par chaque objet.

- ▷ Segmentation ternaire de l'image en 3 classes : *fond*, *forme*, *ombre*, telle qu'illustrée par la figure 4.14 colonne de droite. Si on parvient à mener à bien cette segmentation, on peut ignorer l'*ombre* portée par chaque objet. Malheureusement, les segmentations ainsi obtenues le sont parfois au détriment de la classe *forme*, ou à l'inverse, une partie de l'*ombre* est mal classifiée comme *forme*. Ces deux erreurs sont illustrées sur les colonnes de droite des figures 4.14 et 4.15.

Cette dernière approche est potentiellement la plus riche, mais elle est peu robuste : elle est souvent mise en défaut, et les différents auteurs ont dû intégrer des heuristiques spatiales et temporelles dans leurs méthodes de segmentation. Nous choisissons la première option. Il faut donc modéliser l'éclairement et l'ombre portée par chaque objet.

#### 4.11.1 Intégration de l'éclairement dans le vecteur d'état

Afin de suivre simultanément des objets et leurs conditions d'illumination, le vecteur d'état défini à l'équation (4.1), est enrichi des variables aléatoires encodant l'état de l'illumination. La définition du vecteur d'état (4.1) est donc remplacée par (4.64) :

$$\mathbf{X}_t^n = \{\mathbf{I}_t^n, J_t^n, \mathbf{x}_t^{j,n}\}, \forall j \in \{1, \dots, J_t^n\} \quad (4.64)$$

où  $\mathbf{I}_t^n = \{\xi_t^n, \phi_t^n, \psi_t^n\}$  définit l'illumination candidate de la particule  $n$  à l'instant  $t$ ,  $n \in \{1, \dots, N\}$ , où  $N$  est le nombre de particules. Plus précisément,  $\xi_t^n$  est une variable aléatoire binaire définissant l'état de l'éclairement solaire comme direct ou indirect (soleil caché par un nuage),  $\phi_t^n$  et  $\psi_t^n$  sont des variables aléatoires continues représentant respectivement l'azimut et l'altitude du soleil, comme illustré figure 4.16. Quand la lumière solaire est directe, on suppose les ombres des objets portées sur le sol ou sur d'autres objets.  $J_t^n$  et  $\mathbf{x}_t^{j,n}$  sont tels que définis dans l'équation (4.1).

#### 4.11.2 Mouvements additionnels pour l'éclairement

Dans une application de suivi d'objets, le *FP RJ-MCMC* ne met en œuvre que trois types de mouvements, dont l'ensemble est noté  $\mathcal{M} = \{e, u, s\}$ , où  $e$ ,  $u$  et  $s$  désignent respectivement l'entrée, la mise à jour et la sortie d'un objet. Dans cette section, nous étendons le *FP RJ-MCMC* pour lui permettre de suivre conjointement les objets et la source d'illumination. Pour ce faire, l'ensemble des mouvements est étendu :  $\mathcal{M} = \{e, u, s, es, us, ss\}$ , où les mouvements additionnels  $es$ ,  $us$  et  $ss$  désignent respectivement l'entrée, la mise à jour et la sortie du soleil. Les algorithmes 4.2 et 4.4 doivent évidemment être modifiés pour prendre en compte ces nouveaux mouvements.

- ▷ **Entrée / Sortie du soleil** : Tirer aléatoirement une instance d'état du soleil  $\xi_{t-1}^r$ ,  $r \in \{1, \dots, N\}$  de la chaîne à l'instant  $t - 1$ , et lui proposer de changer, *i.e.* que le soleil entre, générant ainsi des ombres portées des objets sur le sol, ou qu'il sorte (se cache derrière un nuage), faisant disparaître les ombres portées des objets. Quand l'éclairement du soleil est direct, ses propositions de sortie sont conditionnées à sa *vitalité*.
- ▷ **Mise à jour position du soleil** : Tirer aléatoirement une particule  $\mathbf{X}_{t-1}^r$ ,  $r \in \{1, \dots, N\}$  de la chaîne à l'instant  $t - 1$  :  $\{\mathbf{X}_{t-1}^r\}_{r=1}^N$ . Si l'éclairement solaire est supposé direct dans cette particule, appliquer la dynamique solaire à cette instance  $\mathbf{I}_{t-1}^r$ . Le modèle de dynamique solaire est défini par (4.65) et (4.66) :

$$p(\phi^* | \phi_{t-1}^r) = \mathcal{N}(\phi_{t-1}^r, \sigma_\phi^2), \quad (4.65)$$

$$p(\psi^* | \psi_{t-1}^r) = \mathcal{N}(\psi_{t-1}^r, \sigma_\psi^2), \quad (4.66)$$

où  $\sigma_\phi$  et  $\sigma_\psi$  sont respectivement les écarts types d'azimut et d'altitude.



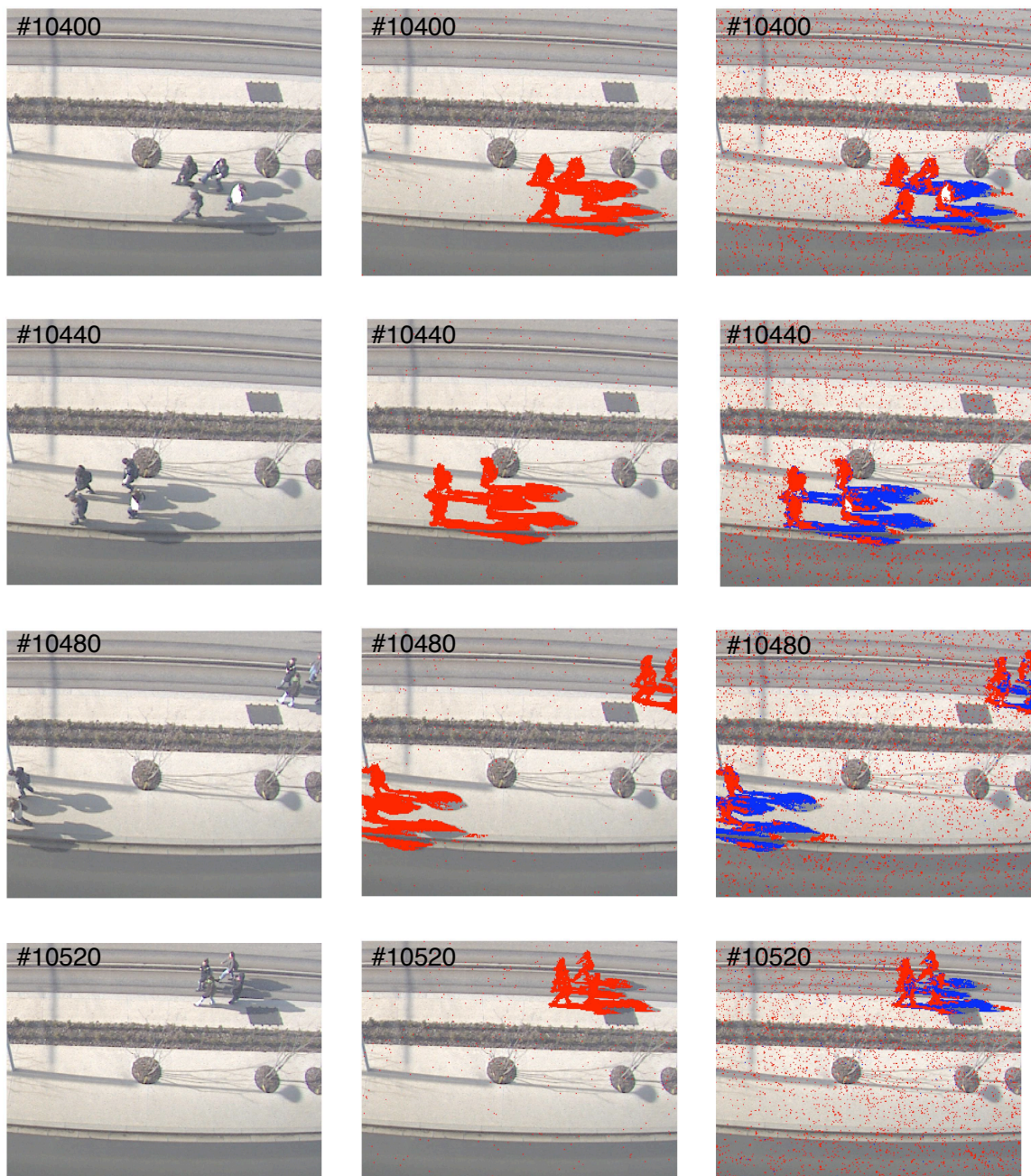


FIGURE 4.14 – Segmentation de l'ombre. Colonne de gauche : images brutes. Colonne milieu : segmentation binaire de la *forme* (en rouge) par l'algorithme  $\Sigma - \Delta$  dans l'espace RGB. Colonne droite : segmentation ternaire de la *forme* (en rouge) et de l'*ombre* (en bleu) par l'algorithme  $\Sigma - \Delta$  dans l'espace RGB puis dans l'espace rgb normalisé.

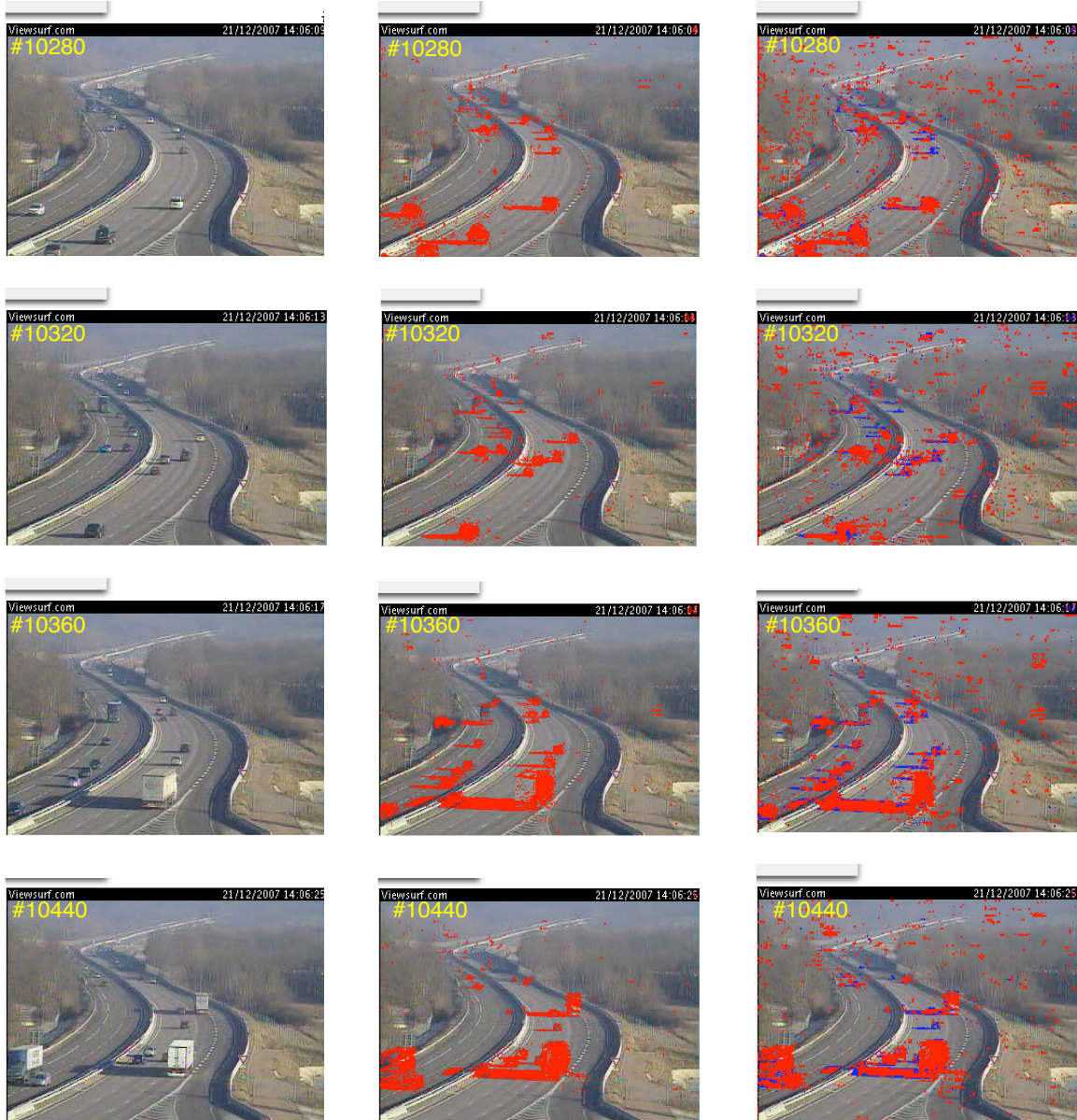


FIGURE 4.15 – Segmentation de l'ombre sur séquence autoroute. Colonne de gauche : images brutes. Colonne milieu : segmentation binaire de la *forme* (en rouge) par l'algorithme  $\Sigma - \Delta$  dans l'espace RGB. Colonne droite : segmentation ternaire de la *forme* (en rouge) et de l'*ombre* (en bleu) par l'algorithme  $\Sigma - \Delta$  dans l'espace RGB puis dans l'espace rgb normalisé.



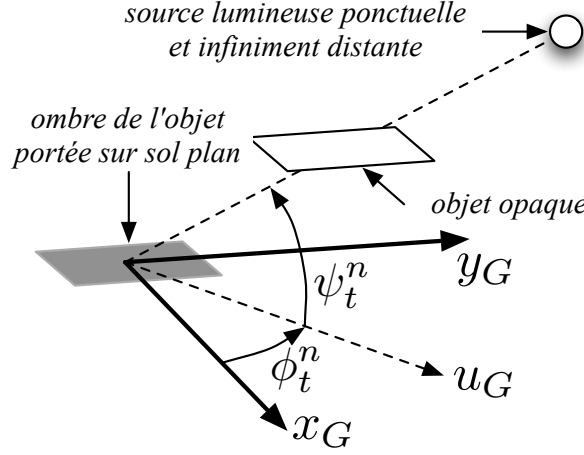


FIGURE 4.16 – Source de lumière candidate définie par la particule  $n$ . La source est supposée ponctuelle et à distance infinie. L'objet est supposé opaque et son ombre supposée portée sur le sol, supposé plan, de plan défini par  $x_G$  et  $y_G$ . Les angles positionnant la source (azimut  $\phi_t^n$ , et altitude  $\psi_t^n$ ) sont relatifs au référentiel local.

#### 4.11.3 Taux d'acceptation d'un mouvement d'éclairement

L'acceptation d'un mouvement d'éclairement est toujours conditionnée par les vraisemblances des configurations globales, défini dans la section 4.4. La différence est que si aucun objet n'est bougé, toutes leurs ombres le sont. La surface de la projection de l'objet bougé, notée  $S_{o,c}$  et requise pour le calcul de l'exposant  $\beta_{j,c}$  par l'équation (4.29), sera ici la somme des surfaces des projections des ombres de tous les objets candidats dans la particule  $\mathbf{X}$ .

**Entrée du soleil :** propose que l'éclairement, précédemment nuageux, devienne ensoleillé, avec la probabilité  $Q(es)$ . Traitant le soleil comme un objet, on peut utiliser l'équation 4.17, ce qui donne un taux d'acceptation défini par 4.67 :

$$\alpha_{es} = \min \left( 1, \frac{w^* Q(ss)}{w_t^{n-1} Q(es)} \right) \quad (4.67)$$

**Sortie du soleil :** propose que l'éclairement, précédemment ensoleillé, devienne nuageux, avec la probabilité  $Q(ss)$ . Traitant le soleil comme un objet, on peut utiliser l'équation 4.19, ce qui donne un taux d'acceptation défini par 4.68 :

$$\alpha_{ss} = \min \left( 1, \frac{w^* Q(es)}{w_t^{n-1} Q(ss)} \right) \quad (4.68)$$

**Mise à jour de position du soleil :** avec la probabilité  $P_{us}$ . Comme pour un objet, on tire une instance de position du soleil  $\phi_{t-1}^r, \psi_{t-1}^r$ . On tire alors  $\phi_t^*, \psi_t^*$  des lois de dynamique du soleil (4.65) et (4.66). Le taux d'acceptation est alors donné par 4.69.

$$\alpha_{us} = \min \left( 1, \frac{w^*}{w_t^{n-1}} \right) \quad (4.69)$$

#### 4.11.4 Mise à jour de la vitalité du soleil.

La *vitalité* du soleil est mise à jour selon le même processus que celle d'un objet, sinon que la région concernée est l'union de toutes les ombres des objets en scène. On remplace l'équation (4.40) par (4.70) :

$$f_t^{s,n} = \frac{1}{|\mathcal{R}_t^{s,n}|} \sum_{g \in \mathcal{R}_t^{s,n}} \overline{\mathbf{I}_F(\mathbf{Z}, g, c)}, \text{ avec } \mathcal{R}_t^{s,n} = \bigcup_{j=1}^{I_t^n} \mathcal{S}_t^{j,n} \quad (4.70)$$

où  $\mathcal{S}_t^{j,n}$  est la région de l'image incluse dans l'enveloppe convexe de la projection de l'ombre portée au sol de l'objet  $j$ . L'incrément de *vitalité* du soleil, noté  $\lambda_t^s$ , est calculé comme l'incrément de *vitalité* d'un objet, sur toute la chaîne de particules (après *Burn-in*, c'est à dire élimination des  $N_B$  premières particules), comme une fonction sigmoïde de  $f_t^{s,n}$  :

$$\lambda_t^s = k_d \cdot \sum_{n=1}^N \frac{e^{-k_r \cdot (f_t^{s,n} - r_f)} - 1}{e^{-k_r \cdot (f_t^{s,n} - r_f)} + 1} \quad (4.71)$$

Le coefficient  $k_d$ , permettant d'ajuster la dynamique de la *vitalité* du soleil, est calculé comme pour celle d'un objet, selon l'équation (4.44), où  $\Lambda_0$  est la *vitalité* initiale du soleil entrant,  $n_s$  indique le nombre d'images pendant lesquelles on maintient la modélisation des ombres. La *vitalité* du soleil est finalement mise à jour pour le prochain instant  $t + 1$ , par l'équation :

$$\Lambda_{t+1}^s = \min(\Lambda_t^s + \lambda_t^s, 1) \quad (4.72)$$

## 4.12 Conclusion

Nous avons présenté dans ce chapitre un système de suivi et de classification visuels multi-objets basé sur le Filtre Particulaire MCMC présenté dans les chapitres précédents. Le système est entièrement intégré au sein d'un Filtre Particulaire *RJ-MCMC*. Nous avons délibérément choisi d'alimenter le filtre avec une observation pauvre, reposant uniquement sur une segmentation binaire *avant-plan / arrière-plan* basée sur un modèle de l'*arrière-plan* mis à jour en ligne à chaque image. Nous avons montré l'intérêt d'utiliser une fonction de vraisemblance relative à la distance objet-caméra. Cette fonction de vraisemblance permet d'améliorer l'efficacité de l'algorithme de Metropolis-Hastings, en asservissant son taux d'acceptation. Nous avons également présenté un potentiel d'interaction entre objets, basé sur une distance anisotrope entre leurs modèles géométriques 3D. Pour améliorer l'efficacité du filtre, nous avons mis en place une procédure de guidage des propositions d'entrée et de sortie des objets, basée sur l'observation, et intégrée au filtre. Pour guider la sortie des objets tout en assurant la cohérence temporelle des objets, nous avons introduit la *vitalité* de chaque objet, un indicateur compilant au cours du temps la pertinence de chaque objet candidat, par rapport à l'observation. Cette variable est mise à jour à chaque pas temporel, et permet de régler en fonction du contexte, la durée de survie des objets candidats aux occultations. Pour réaliser simultanément la catégorisation et le suivi, la catégorie de l'objet a été considérée comme une variable aléatoire discrète, intégrée dans l'état de chaque objet, et suivie par le filtre au même titre que les autres variables. Cette approche conjointe est également d'un grand intérêt car elle permet d'exploiter la dynamique des objets au même titre que leurs formes pour les catégoriser. Cette fonctionnalité peut être utilisée pour améliorer le suivi et la catégorisation sur des routes sinueuses ou des zones urbaines, où la dynamique du véhicule diffère d'une classe à l'autre. En outre, elle pourrait aider à classer les cyclistes et les piétons grâce à la différence de leurs dynamiques. Nous avons enfin modélisé l'ombre portée au sol par les objets présents, générée par le soleil, source lumineuse supposée unique, ponctuelle, et infiniment distante. Pour ce faire, nous avons introduit des variables aléatoires d'illumination dans l'état de la configuration globale de la scène, et introduit le suivi de ces variables au sein du filtre. La source d'illumination est ainsi suivie par le filtre, au même titre que les objets.

### Récapitulatif des mouvements mis en œuvre dans le *FP RJ-MCMC*.

- ▷ Mise à jour d'un objet. Mouvement auto-réversible.
- ▷ Entrée / Sortie d'un objet. Paire de mouvements inverse l'un de l'autre.
- ▷ Mise à jour du soleil. Mouvement auto-réversible.
- ▷ Entrée / Sortie du soleil. Paire de mouvements inverse l'un de l'autre.

---

## EXPÉRIMENTATIONS ET RÉSULTATS

---

Ce chapitre décrit et analyse une campagne d'expérimentations visant à évaluer les performances du système de suivi visuel multi-objet, présenté dans ce manuscrit. Dans la première campagne d'expérimentation, nous comparons la capacité de trois configurations du *FP RJ-MCMC* à maintenir l'identité des objets à travers des périodes d'invisibilité de ces objets. Les résultats montrent l'intérêt de la *vitalité* introduite au chapitre 4. Dans une deuxième campagne, nous menons des expérimentations de suivi réel dans une grande variété de situations, sur piétons, motos, véhicules légers, camionnettes et poids lourds, afin de montrer que le traqueur *FP RJ-*

*MCMC* reste générique, c'est à dire qu'il peut suivre n'importe quel type d'objet opaque et rigide tel qu'un véhicule, ou articulé mais dont la silhouette ne varie pas trop, tel qu'un piéton. Des critères sont proposés pour quantifier les performances de suivi et de classification des objets. La catégorisation des objets intégrée au suivi, et la modélisation de l'éclairement ont été proposées au chapitre 4. Les résultats d'expériences du présent chapitre montrent que la modélisation de l'ombre et la catégorisation d'objets améliorent toutes deux indépendamment les performances. Les meilleurs résultats sont obtenus lorsque les deux sont activées.

---



## 5.1 Introduction

Plusieurs expérimentations ont été menées afin d'évaluer les performances de notre traqueur *FP RJ-MCMC*. La première campagne d'expérimentations présentée dans ce chapitre porte sur l'effet du recours à la *vitalité* des objets, pour guider les propositions de sortie. Nous avons mis en évidence dans le chapitre 4 la difficulté à assurer la cohérence temporelle des identités respectives des objets dans le cas de nos applications de suivi de piétons et de véhicules, où les apparences des objets sont souvent très proches. Khan *et al.*, confrontés à la même difficulté, ont proposé l'adjonction de la paire de mouvements *stay* et *leave* (KBD05), dans le but d'améliorer la cohérence temporelle du nombre d'objets présents dans le champ de vision et de leurs identités respectives. Nous avons proposé dans le chapitre 4 d'affecter à chaque objet une variable déterministe additionnelle, la *vitalité*, mise à jour sur la base des observations. Elle permet de rendre réglable le nombre d'observations nulles auquel un objet survivra et conservera son identité. Les performances respectives de la solution de Khan *et al.* (mouvements *stay* et *leave*) et de la nôtre (propositions de sortie de chaque objet guidées par sa *vitalité*), vis à vis de la cohérence temporelle des identités des objets suivis, sont comparées dans ce chapitre. Ces expérimentations sont menées sur données de synthèse.

La suite des expérimentations vise à valider la capacité de notre traqueur *FP RJ-MCMC* à suivre et catégoriser un nombre variable d'objets multi-objets en utilisant aussi peu de données *a priori* que possible. Cette contrainte a été posée pour que cet outil reste générique, donc applicable à des besoins variés. Pour valider cette « généralité », nous présentons dans ce chapitre des expériences de suivi sur différents scénarios, impliquant des piétons, des motos, des véhicules légers, des camionnettes et poids lourds. Les fonctionnalités suivantes sont évaluées :

- ▷ précision du suivi,
- ▷ résistance aux variations des conditions d'illuminations.

Ce chapitre est organisé ainsi : en section 5.2, nous présentons les matériaux vidéos utilisés pour la suite des expérimentations, et les moyens d'évaluation des performances. En section 5.3.1, nous comparons les capacités respectives de la méthode de Khan *et al.* et de la nôtre, à maintenir la cohérence temporelle des identités des objets suivis. Les performances obtenues sont ensuite exposées dans les sections 5.4 à 5.8.2.

## 5.2 Matériau Vidéo et Méthodologie

**Cohérence Temporelle des Identités.** Les expérimentations de cohérence temporelle des identités visent à comparer l'effet de la *vitalité* et de la paire de mouvements *stay-leave* proposée dans (KBD05). Pour ce faire, nous avons généré des séquences vidéo de synthèse spécifiques. Ces expérimentations sont menées en 2D, sur des objets rectangulaires de  $21 \times 21$  pixels, se déplaçant horizontalement de la gauche vers la droite, à une vitesse constante arbitrairement fixée à 2 pixels par image, sur un champ de  $240 \times 240$  pixels. Compte tenu d'une marge de 20 pixels sur les bords de l'image, la zone de suivi est un champ de  $200 \times 200$  pixels. Chaque objet est donc présent dans cette zone pendant 100 images. Un nouvel objet entre sur la gauche du champ toutes les 8 images, comme illustré par la figure 5.1, où un nouvel objet apparaît dans le champ aux images #10008, #10016, #10024, etc... (la numérotation commence à #10000). Il quittera le champ lorsqu'il aura atteint le bord droit de l'image. On a donc en moyenne 11.5 objets simultanément perceptibles. La séquence test comporte 700 images, pendant lesquelles 75 objets traversent la zone de suivi. Pour simuler l'éventualité qu'un objet reste non observable pendant plusieurs images consécutives, chaque objet peut disparaître complètement de l'observation à chaque image avec une probabilité  $P_i$  arbitrairement fixée à  $P_i = 0.02$ . Lorsque cela se produit, il reste invisible pendant une durée de  $I$  images consécutives, nombre que nous faisons varier de 1 à 10 images au cours des expérimentations. Sur la figure 5.1, cette durée vaut  $I = 5$  images. Le 1<sup>er</sup> objet apparu est ainsi invisible à l'image #10040, le 3<sup>ème</sup> objet est invisible à l'image #10048. La ligne du bas détaille ces phases de disparition : le 7<sup>ème</sup> objet disparaît à l'image #10066, pour être à nouveau visible à l'image #10071. Le but de cette expérimentation est d'établir la probabilité avec laquelle le traqueur lui réaffecte l'identité qu'il portait avant de disparaître. Les résultats sont présentés en section 5.3.1. En complément au matériau de synthèse décrit ci-dessus, deux séquences réelles de suivi de piétons ont été analysées :

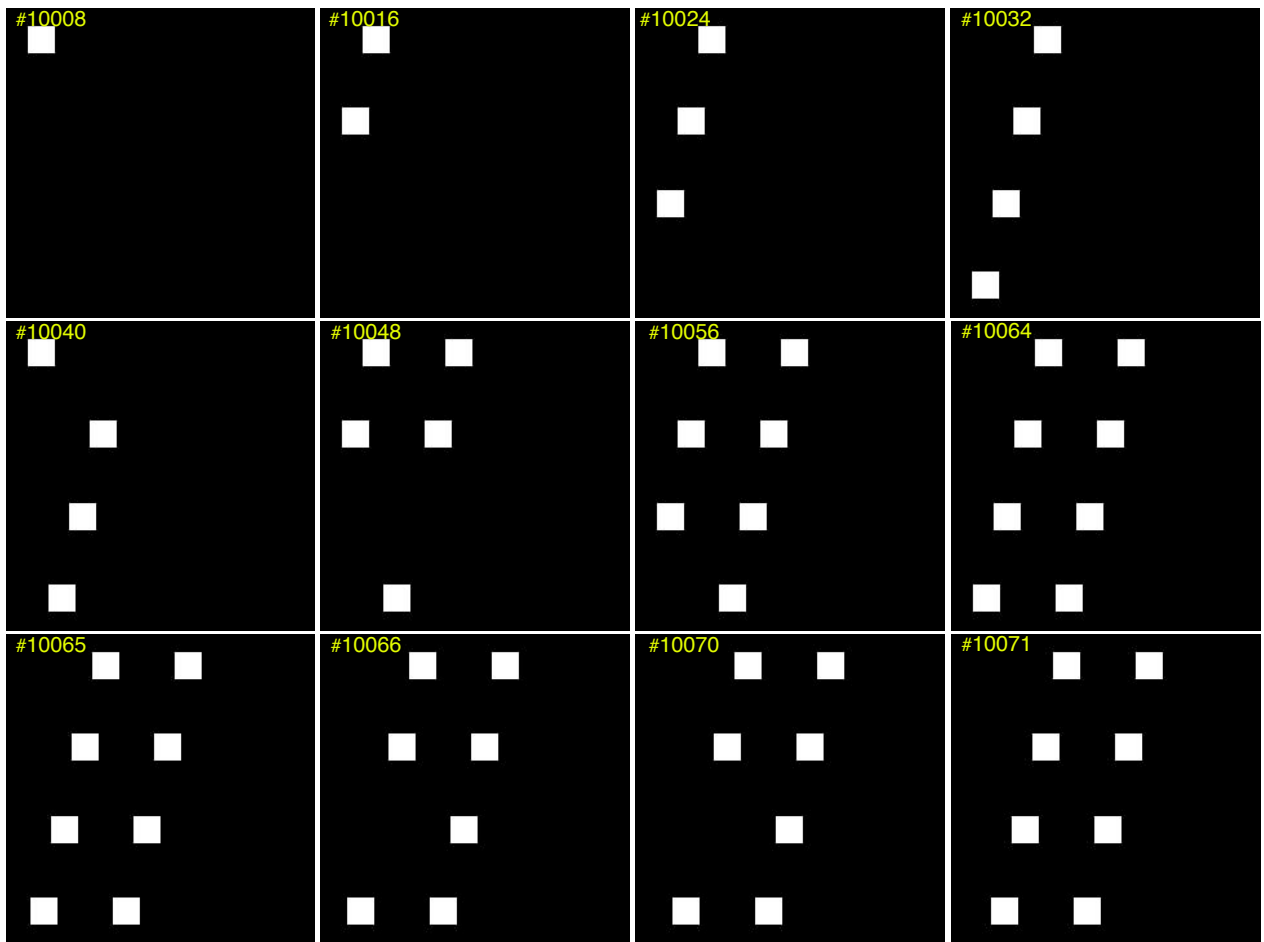


FIGURE 5.1 – Extraits de la séquence de synthèse dédiée à l'évaluation du maintien de l'identité des objets. Objets identiques de  $21 \times 21$  pixels, se déplaçant horizontalement de la gauche vers la droite, à une vitesse constante arbitrairement fixée à 2 pixels par image, sur un champ de  $240 \times 240$  pixels. Un nouvel objet entre sur la gauche du champ toutes les 8 images.

- ▷ Suivi de piéton en mono-vision sur un espace très réduit ( $3.5 \times 3.5$  mètres), pour illustrer qualitativement les limites de l'identification des objets par notre traqueur. Les résultats sont présentés en section 5.3.2.
- ▷ Suivi de piéton en multi-vues. Il s'agit d'analyser les apports d'une observation multi-vues, pour maintenir les identités des objets malgré les occultations. Les résultats sont présentés en section 5.3.3.

**Suivi et classification conjoints sous illumination variable.** La deuxième partie des expérimentations permet d'évaluer les performances de notre traqueur *FP RJ-MCMC* sur des séquences vidéos réelles, aussi bien que sur des séquences de synthèse. Ces données sont échantillonnées sur différents scénarios appartenant à deux domaines d'applications : le suivi de piétons et le suivi de véhicules dans le trafic routier. Dans le premier cas, la classification n'est pas activée. Dans le cas du suivi de véhicule dans le trafic, le système sera expérimenté avec et sans classification. Comme notre traqueur doit pouvoir fonctionner en conditions dégradées, les séquences réelles sont tirées de webcams de surveillance de voirie publique, non calibrées, de basse résolution :  $320 \times 240$  pixels, et fortement compressées. Les matrices de projection sont approximées à la main. Les objets cibles situés à l'intérieur de la zone de suivi, sont suivis et classifiés par le système. La zone de suivi est définie par l'utilisateur dans le repère absolu, et représentée sur les images par les contours en trait vert : voir figures 5.10 page 117, 5.11 page 118 et 5.15 page 123. Sur ces figures, les lignes vertes définissent les limites de la zone de suivi. Lorsqu'un nouvel objet entre dans ce domaine, il est étiqueté avec un identifiant unique, et la projection de son estimation est tracée.

**Séquences d'expérimentation pour le suivi et la classification conjoints sous illumination variable.** Le suivi a été mené sur plusieurs scénarios :

1. Suivi d'une moto instrumentée. Il s'agit d'évaluer la précision du suivi de trajectoire. La moto est équipée d'un GPS RTK (précision centimétrique) pour fournir une vérité terrain. Une caméra couleur  $480 \times 640$  vise l'entrée de courbe, et une autre la sortie de courbe, comme indiqué sur la figure 5.9 page 115. Les résultats obtenus sont exposés en section 5.4.
2. Suivi de piétons sous illumination variable, caméra couleur  $480 \times 640$ . Ces expérimentations visent à évaluer la capacité du système à suivre un nombre variable d'objets sous une illumination variable, notamment vis à vis de la génération d'ombres portées et de la direction du soleil. Les résultats obtenus sont exposés en section 5.5.
3. Suivi et classification de véhicules (véhicules légers, camionnettes, poids-lourds) sur autoroute, sous illumination choisie pour ne pas poser de problème d'ombre. Ces expérimentations, précédemment présentées dans (BCR09c), visent à valider le suivi et classification simultanés d'objets, ainsi qu'à évaluer l'intérêt du potentiel d'interaction entre objets présentée dans la section 4.8. Le site retenu est la sortie d'une barrière de péage, capturée avec une webcam de faible résolution ( $320 \times 240$ ), faible fréquence (15 images par seconde) (voir figure 5.15 page 123). Les résultats obtenus sont exposés en section 5.6.
4. Suivi et classification de véhicules (véhicules légers, camionnettes, poids-lourds) sur autoroute sous illumination variable, notamment vis à vis de la génération d'ombres portées. Ces expérimentations visent à comparer les performances du système, en fonction de l'activation de la classification et de la modélisation de l'illumination. Les résultats obtenus sont exposés en section 5.7.
5. Suivi et classification de véhicules (voitures, motos et camions) sur route sinueuse. L'objectif de ces expérimentations est de valider la capacité du système à réaliser le suivi sur différents types de trajectoires. Les résultats obtenus sont exposés en section 5.8.2.

**Critères de performance de suivi et classification.** Définissons d'abord les conditions dans lesquelles un objet cible sera considéré comme correctement suivi :

- ▷ un objet est estimé présent à l'instant  $t$ , associé à cet objet cible,

- ▷ la distance entre l'objet estimé et l'objet cible n'excède pas la plus grande dimension au sol de l'objet estimé.

Les performances de suivi et classification conjoints sous illumination variable sont alors évaluées grâce à quatre critères :

- ▷ Le taux de suivi  $\theta_T = \frac{1}{J_t} \sum_{t,j} \delta_T(t, j)$  où  $\delta_T(t, j) = 1$  si l'objet cible  $j$  est correctement suivi à l'instant  $t$ , sinon 0.  $J_t = \sum_t j_t$ , où  $j_t$  est le nombre d'objets cibles dans la zone de suivi à l'instant  $t$ .
- ▷ Le taux de classification  $\theta_C = \frac{1}{J_t} \sum_{t,j} \delta_C(t, j)$  où  $\delta_C(t, j) = 1$  si l'objet cible  $j$  est correctement classifié à l'instant  $t$ , sinon 0.
- ▷ Le taux de fantômes  $\theta_G = \frac{1}{J_t} \sum_{t,j} \delta_G(t, j)$  où  $\delta_G(t, j)$  est le nombre d'objets candidats fantômes, c'est à dire ne correspondant à aucun objet cible.
- ▷ Erreur moyenne de localisation  $\epsilon_T = \frac{1}{J_t} \sum_{t,j} \sqrt{\delta_p^T(t, j) \cdot \delta_p(t, j)}$ , où  $\delta_p(t, j) = \mathbf{p}_v(t, j) - \mathbf{p}_{truth}(t, j)$ , où  $\mathbf{p}_v(t, j)$  est la position de l'objet  $j$  à l'instant  $t$  estimée par le système de suivi, et  $\mathbf{p}_{truth}(t, j)$  est la position réelle « vérité terrain » de l'objet  $j$  à l'instant  $t$ .

**Implémentation.** Nous utilisons l'extension parallélisée de l'algorithme MCMC PF, sur une unité de traitement à deux cœurs. A chaque itération, deux propositions de configurations multi-objets sont générées en parallèle, et les vraisemblances de l'observation, conditionnées à ces configurations, sont calculées en parallèle, par des *threads* fournis par les Bibliothèques *Boost C++*<sup>1</sup>. Le code est écrit pour utiliser la Bibliothèque *NT*<sup>2</sup> C++<sup>2</sup>. Nous utilisons un PC avec processeur *Intel E6850 Core 2 Duo* cadencé à 3GHz, avec 4Go de RAM, sous *Linux*. Toutes les expérimentations présentée ici ont été menées à la fréquence du temps réel vidéo, *i.e.* 25 images par seconde, en mono-vision, sur des images de  $320 \times 240$ . Pour les expérimentations de cohérence temporelle des identités, le nombre de particules est fixé à  $N = 100$ , ce qui est peu pour suivre une moyenne de 11.5 objets simultanément présents, mais suffisant dans ce scénario  $2D$  où les objets n'interagissent pas. Pour toutes les autres expérimentations, le nombre de particules est fixé à  $N = 200$ .

### 5.3 Cohérence Temporelle des Identités

Les propositions d'*entrée* et de *sortie* d'objets présentées en section 4.3.1 n'assurent pas la cohérence temporelle de présence des objets. Bien au contraire, ces mouvements vont proposer qu'un objet candidat sorte, et si cette sortie est acceptée, un nouvel objet candidat auquel le filtre proposera peut-être d'entrer ultérieurement, portera une nouvelle identité. Pour donner de la cohérence temporelle aux identités des objets, Khan *et al.* mettent en œuvre une deuxième paire de mouvements, dénommés *stay* et *leave* (KBD05), que nous avons codés de la façon suivante, pour tester leur impact sur la cohérence temporelle des identités :

- ▷ **Stay** : ce mouvement propose une nouvelle configuration jointe  $\mathbf{X}_t^* = \{\mathbf{X}_t^{n-1}, \mathbf{x}_t^{j*}\}$ , ajoutant un objet  $\mathbf{x}_t^{j*}$  généré en appliquant la loi dynamique  $p(\mathbf{x}_t | \mathbf{x}_{t-1})$  à l'objet  $\mathbf{x}_{t-1}^{j,r}$  contenu dans au moins une des particules de la chaîne à l'instant  $t - 1$ , mais pas dans  $\mathbf{X}_t^{n-1}$ . La figure 5.2 illustre ce mouvement. Le processus de recherche « saute » vers un sous-espace de plus grande dimension.
- ▷ **Leave** : ce mouvement propose une nouvelle configuration jointe excluant l'objet  $j$  de la particule précédente s'il n'est présent dans aucune des particules de la chaîne à l'instant précédent  $t - 1$  :  $\mathbf{X}_t^* = \{\mathbf{X}_t^{n-1} \setminus \mathbf{x}_t^{j,n-1}\}$ ,  $j \in \{1, \dots, J_t^{n-1}\}$ , où  $J_t^{n-1}$  est le nombre d'objets dans l'hypothèse  $\mathbf{X}_t^{n-1}$ , et  $\{\mathcal{S} \setminus \mathcal{E}\}$  est l'ensemble  $\mathcal{S}$  privé de l'élément  $\mathcal{E}$ . La figure 5.3 illustre ce mouvement. Le processus de recherche « saute » vers un sous-espace de plus petite dimension.

<sup>1</sup><http://www.boost.org>

<sup>2</sup>Numerical Template Toolbox. <http://nt2.sourceforge.net>

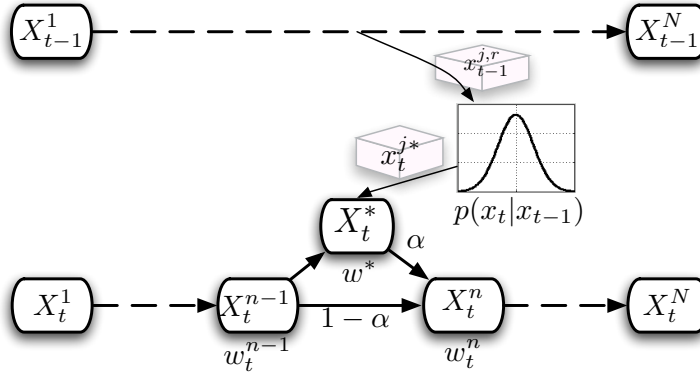


FIGURE 5.2 – Mouvement *stay* d'un objet à l'itération  $n$  du filtre *FP MCMC*. La proposition  $\mathbf{X}_t^*$  est construite en ajoutant à la particule  $\mathbf{X}_t^{n-1}$  un objet  $\mathbf{x}_t^{j,*}$ , généré en appliquant la loi dynamique  $p(\mathbf{x}_t | \mathbf{x}_{t-1})$  à l'objet  $\mathbf{x}_{t-1}^{j,r}$  qu'elle ne contient pas, mais qui était décrit par au moins une des particules de la chaîne à l'instant  $t - 1$ .

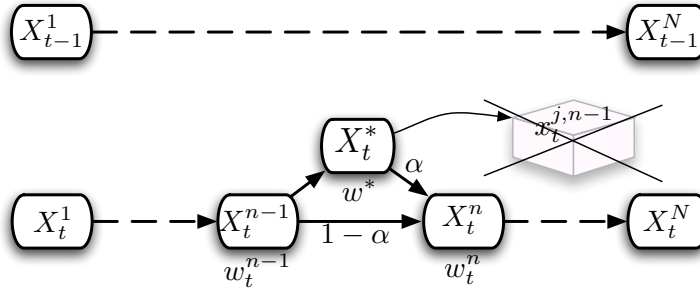


FIGURE 5.3 – Mouvement *leave* à l'itération  $n$ , de l'objet  $\mathbf{x}_t^{j,n-1}$  présent dans le maillon précédent  $\mathbf{X}_t^{n-1}$  de la chaîne courante à  $t$ , mais présent dans aucune des particules de la chaîne à l'instant  $t - 1$ . La proposition  $\mathbf{X}_t^*$  est construite en supprimant cet objet de la particule  $\mathbf{X}_t^{n-1}$ .

Face à la paire de mouvements *stay* et *leave* proposée par Khan *et al.*, nous avons proposé dans le chapitre 4 d'affecter à chaque objet une variable additionnelle, la *vitalité*, mise à jour sur la base des observations. Les performances respectives de la solution de Khan *et al.* sont comparées à la nôtre dans cette section, vis à vis de la cohérence temporelle des identités des objets suivis.

### 5.3.1 Suivi 2D sur données de synthèse

Pour ces expérimentations, le nombre de particules du *FP RJ-MCMC* est fixé à  $N = 100$ . La dynamique est définie par le modèle à vitesse constante selon les équations (4.3), (4.4) et (4.5), avec  $\sigma_m = 0.5$  pixels/image et  $\sigma_a = 0.1$  rad. La forme de l'objet est mise à jour selon (4.6), avec  $\sigma_s = 0.5$  pixels/image. Trois configurations du *FP RJ-MCMC* sont comparées :

- ▷ *FP RJ-MCMC* seulement muni de la paire de mouvements *entrée-sortie*, sans la paire de mouvements *stay-leave*, ni *vitalité*.
- ▷ *FP RJ-MCMC* muni de la paire de mouvements *entrée-sortie*, et de la paire de mouvements *stay-leave*, comme proposé par Khan *et al.* (KBD05), et sans *vitalité*.
- ▷ *FP RJ-MCMC* muni de la paire de mouvements *entrée-sortie* et de la *vitalité*. Dans ces expérimentations, le paramètre  $n_s$  (cf. section 4.6.3) déterminant la dynamique de *vitalité* est réglé à  $n_s = 10$  images. Rappelons que ce paramètre définit le nombre d'images d'invisibilité totale auquel un objet de *vitalité* initialement maximale peut survivre. Les autres paramètres sont conformes aux données de la section 4.6.3.

TABLE 5.1 – Nombre de changements d’identité d’objets résultant d’une absence totale d’observation de l’objet pendant  $I$  images.

$I$ (nombre d’images d’invisibilité)	1	2	3	4	5	10
<i>FP RJ-MCMC</i>	56	83	89	92	107	120
<i>FP RJ-MCMC stay-leave</i>	49	69	84	89	102	106
<i>FP RJ-MCMC Vitalité</i>	4	6	11	14	25	110

La gestion des disparitions provisoires d’objets par ces trois configurations est illustrée par la figure 5.4 page 108, où le nombre d’images d’invisibilité consécutives vaut  $I = 5$ . L’objet désigné par la flèche jaune à l’image #10065 (identifié comme #9 par les 2 premières configurations du traqueur, par #7 par le traqueur avec *vitalité*), devient invisible à l’image #10066. Les deux configurations sans *vitalité* (haut et milieu) estiment que cet objet est sorti dès l’image #10067, tandis que la configuration avec *vitalité* (bas) maintient l’existence de cet objet, mais décrémente sa *vitalité* tant qu’il n’est pas observable. Ceci permet au *FP RJ-MCMC* de recoller l’objet #7 à sa cible dès l’image #10071, où elle redevient observable, donc de maintenir son identité. Cette dynamique de la *vitalité* est réglée en fonction de la durée des périodes d’invisibilité des objets, comme décrit au chapitre 4. Pendant ce temps, les deux autres configurations ont perdu l’identité initiale de l’objet, et instancient un nouvel objet sur cette cible, respectivement identifiée comme #11 et #12. La figure 5.5 présente ce comportement sur toute la séquence, mettant en évidence la cohérence temporelle d’identité des objets apportée par la *vitalité*. Les résultats sont résumés dans le tableau 5.1, montrant que les configurations sans *vitalité* sont incapables d’assurer cette cohérence, même pour de faibles durées d’invisibilité. Ce résultat n’est pas surprenant pour la première configuration du traqueur (sans mouvements *stay-leave* ni *vitalité*), car le filtre ne propage alors pas les différents états joints multi-objets de la chaîne entre l’instant  $t - 1$  et  $t$  : il se contente de propager les états disjoints de chaque objet. Ce résultat est par contre décevant pour la deuxième configuration du traqueur (avec mouvements *stay-leave*), car le filtre propage alors les différents états joints multi-objets de la chaîne entre l’instant  $t - 1$  et  $t$  : on pouvait alors s’attendre à ce que ces mouvements assurent une inertie des états joints composant la chaîne. Ce mécanisme est certainement efficace dans le cas de la fonction d’observation utilisée dans (KBD05), s’appuyant sur des scores de détection d’objets. Par contre dans le scénario choisi ici, mettant en œuvre une invisibilité totale, ce mécanisme est mis en défaut. Dans le cas de l’invisibilité totale et avec la fonction d’observation que nous utilisons, seule la méthode munie de la *vitalité* permet d’assurer la cohérence temporelle. Ce mécanisme présente une limite, illustrée par la dernière colonne du tableau, où le nombre d’images d’invisibilité consécutives vaut  $I = 10$ . Comme le paramètre de dynamique de *vitalité*  $n_s$  est réglé sur cette valeur, le *FP RJ-MCMC* propose fréquemment la sortie de l’objet au terme de sa période d’invisibilité. La *vitalité* n’a alors plus d’effet significatif, comme le montrent les résultats du tableau 5.1. Rappelons que la résistance à des périodes d’invisibilité forte ou totale est rendue nécessaire du fait des occultations pouvant se produire, causées soit par un autre objet, soit par un élément occultant du fond, derrière lequel l’objet passe, tel qu’un arbre, un panneau, un pont...

### 5.3.2 Suivi 3D de piétons en mono-vision sur espace réduit

Un espace réduit ( $3.5 \times 3.5$  mètres) a été choisi pour mettre en évidence les limites de l’identification des objets par notre traqueur. Quelques images-clés de la séquence sont données figure 5.6 page 111. Le suivi des deux cibles est convenable jusqu’à l’image #094, les deux cibles étant identifiées #7 et #9. A l’image #065, la cible #9 sort de l’espace de traque, et lorsqu’elle y rentre à l’image #094, elle reçoit un nouvel identifiant, ce qui est normal puisque nous n’avons implanté aucun mécanisme de reconnaissance de l’objet vu précédemment. Plus gênant : à l’image #101 cet objet sort à nouveau car il est à la fois occulté par #7, et au bord de l’espace de traque. La cible, qui n’est plus suivie, attire à elle l’objet bleu et le subtilise à l’autre cible, sur laquelle le traqueur initialise un nouvel objet à l’image #136. L’espace réduit (les cibles sont très souvent proches des bords) combiné aux occultations, exacerbe ce dysfonctionnement, si bien qu’à l’image #359 les deux cibles

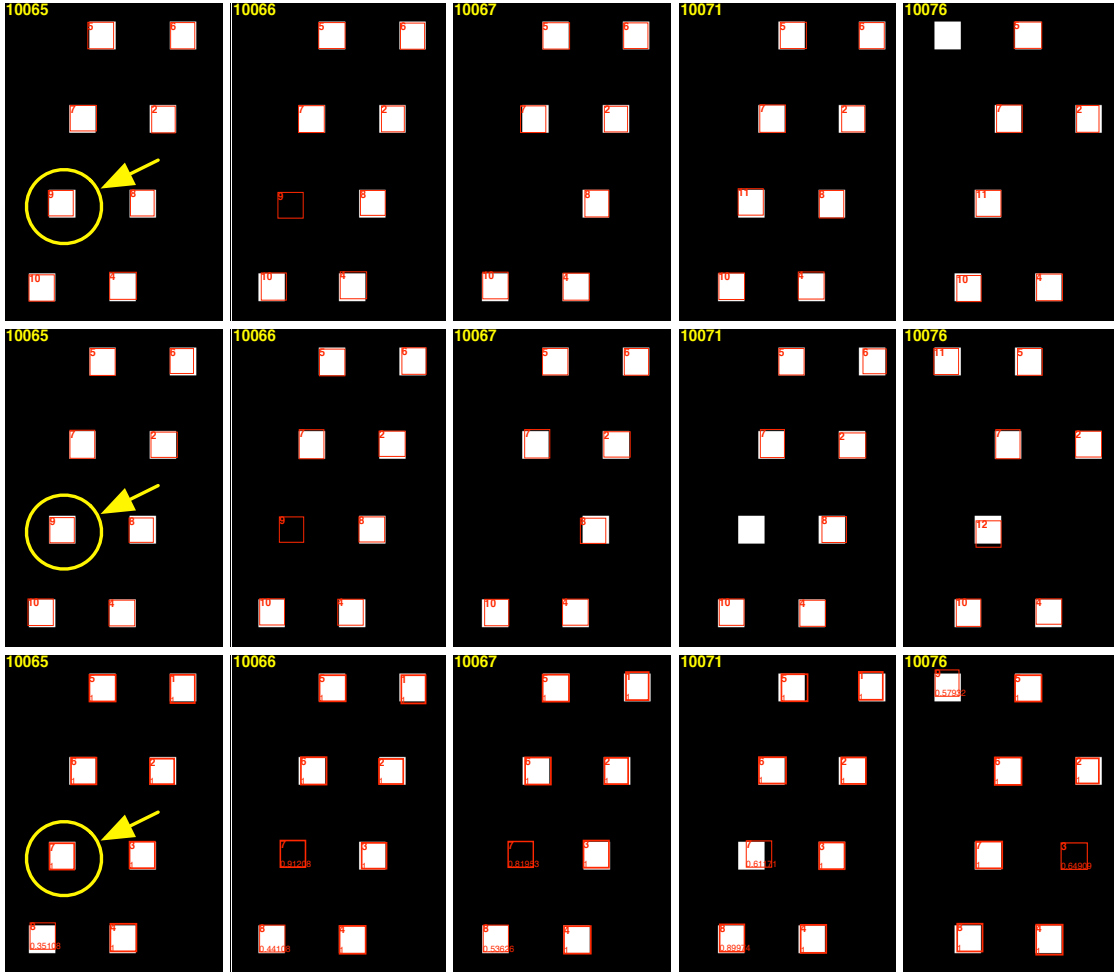


FIGURE 5.4 – Cohérence temporelle des identifiants après une période d’invisibilité de l’objet désigné par la flèche jaune pendant  $I = 5$  images : il est totalement invisible de l’image #10066 à l’image #10070. L’identité est inscrite en haut à gauche de chaque objet estimé par le traqueur. Haut : *FP RJ-MCMC* avec la paire de mouvements *entrée-sortie* mais sans la paire de mouvement *stay-leave*. L’identité de l’objet à l’image #10065 est #9. Lorsqu’il redevient visible, il reçoit une nouvelle identité : #11. Milieu : *FP RJ-MCMC* avec les deux paires de mouvements *entrée-sortie* et *stay-leave*, comme proposé par Khan *et al.* . L’identité de l’objet à l’image #10065 est #9. Lorsqu’il redevient visible, il reçoit une nouvelle identité : #12. Bas : *FP RJ-MCMC* avec la paire de mouvements *entrée-sortie* et la *vitalité*. L’identité de l’objet est maintenue à #7 pendant toute la période. La *vitalité* est inscrite en bas de chaque objet estimé par le *FP RJ-MCMC*.



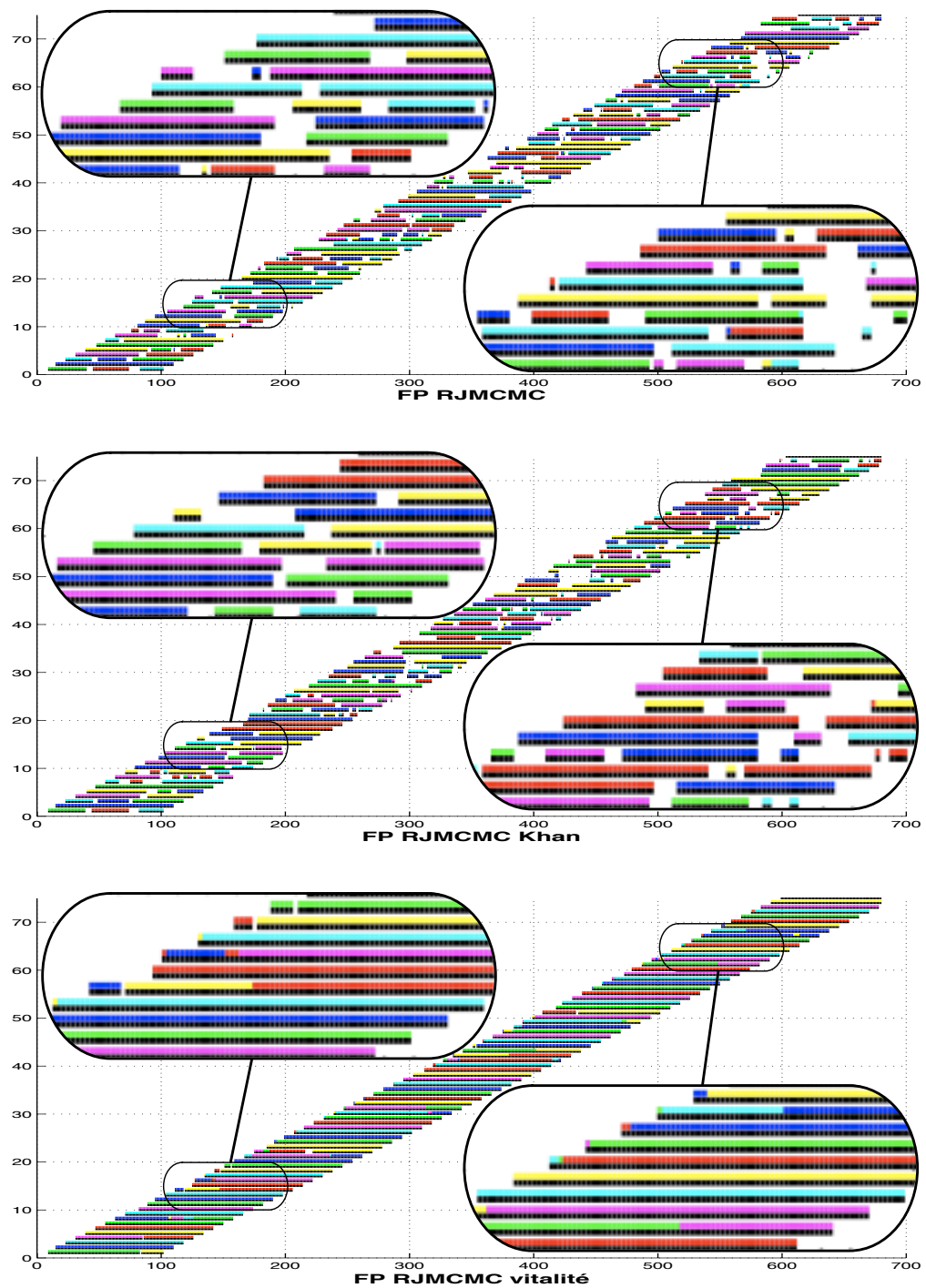


FIGURE 5.5 – Chronogramme de présence des objets sur la séquence, pour une durée d’invisibilité des objets  $I = 5$  images. Temps en abscisse. La présence d’une cible est représentée par une barre noire. L’ordonnée de la barre donne l’ordre d’apparition de la cible. Lorsqu’une cible est suivie, la barre noire est surmontée d’une barre dont la couleur représente l’identité de l’objet estimée par le traqueur, sur un cycle de 6 couleurs (rouge vert bleu, cyan, magenta, jaune). Haut : *FP RJ-MCMC* avec la paire de mouvements *entrée-sortie* mais sans la paire de mouvement *stay-leave*. Milieu : *FP RJ-MCMC* avec les deux paires de mouvements *entrée-sortie* et *stay-leave*, comme proposé par Khan *et al.* . Bas : *FP RJ-MCMC* avec la paire de mouvements *entrée-sortie* et la *vitalité*.



portent les identités #24 et #27 !

### 5.3.3 Suivi 3D de piétons en multi-vues

Cette séquence a pour objectif d'illustrer qualitativement l'apport mais aussi les limitations liés à l'exploitation d'une observation multi-vues (deux vues dans notre exemple). Quelques images-clés de la séquence sont données sur les figures 5.7 page 112 et 5.8 page 113.

**Avantages :** il est clairement illustré par les figures 5.7 et 5.8, où le traitement des deux vues permet de lever les ambiguïtés causées par les occultations mutuelles. Si la configuration illustrée à l'image #00102 ne pose pas de problème, il n'en va pas de même à l'image #00117, caméra #1, ni à l'image #00128, caméra #2, où les piétons suivis en orange et magenta s'occultent presque totalement. En monovision, pour récupérer les identités correctes après occultation, il faudrait remplir au moins l'une des conditions suivantes : 1) appliquer un modèle dynamique très contraint, ou 2) disposer d'un descripteur d'apparence suffisamment discriminant pour ré-associer les objets à leur cible après occultation. Si la première condition est envisageable pour du suivi d'objets à la dynamique très contrainte tels que des véhicules, elles conduisent à de nombreux échanges d'identité dans le cas du suivi de piétons, dont la dynamique est peu contrainte. La deuxième condition est difficile à mettre en œuvre car un descripteur d'apparence à la fois robuste aux changements de pose et d'illumination, et suffisamment discriminant, est un vaste programme. Comme nous ne remplissons aucune de ces deux conditions, seule l'observation multi-vues permet au suivi de maintenir correctement les identités après ces occultations, et ces deux objets se retrouvent correctement suivis et identifiés à l'image #00157.

**Limitations :** les figures 5.7 page 112 et 5.8 page 113, montrent aussi que le traqueur, bien que disposant de l'observation délivrée par deux caméras, est mis en défaut dans le suivi du piéton cible vêtu de noir, (suivi par l'objet de couleur mauve). Cet objet se cale sur sa cible lorsqu'elle entre dans l'espace de traque en fond de scène à l'image #00102, et la suit correctement jusqu'à l'image #00128, à partir de laquelle il va être attiré par la cible en veste verte, non encore suivie car en train d'entrer dans la zone, et qui l'occulte dans la caméra #2. A l'image #00157, le dysfonctionnement qui en résulte est double : 1) l'objet mauve suit une cible « fantôme », 2) aucune des deux cibles impliquées n'est correctement suivie. Les deux cibles sont enfin correctement estimées à l'image #00190, mais la cible en veste verte a subtilisé l'identité de la cible vêtue de noir. Ce dysfonctionnement est causé par la non détection de la cible à veste verte (entrant dans l'espace de traque à l'image #00128), non détection dont la mauvaise segmentation *fond / forme* sur la vue #1 est en partie responsable, et très marquée notamment à l'image #00157. Il y a donc une marge de progrès sur la segmentation *fond / forme*. Enfin, l'image #00697 montre que le traqueur s'en sort bien dans le reste de la séquence, malgré des occultations fortes et fréquentes telles que dans la vue #1 entre les piétons suivis en orange et magenta.

**Contraintes liées au multi-vues.** Une installation multi-vues impose des contraintes supplémentaires :

- ▷ installation matérielle et calibration des caméras,
- ▷ synchronisation des caméras,
- ▷ temps de calcul : l'opération la plus coûteuse en calculs est l'évaluation de la vraisemblance de l'observation pour chaque configuration multi-objets. Ces coûts sont évidemment multipliés par le nombre de caméras ! Notre priorité étant le temps réel, c'est la raison pour laquelle nous avons favorisé les expérimentations en mono-vision.

## 5.4 Suivi de moto seule avec vérité GPS

Certaines applications nécessitent une bonne précision, de l'ordre de quelques centimètres pour du suivi de véhicules. C'est le cas des études de trajectoires de véhicules. Cette série d'expérimentations a pour objectif la

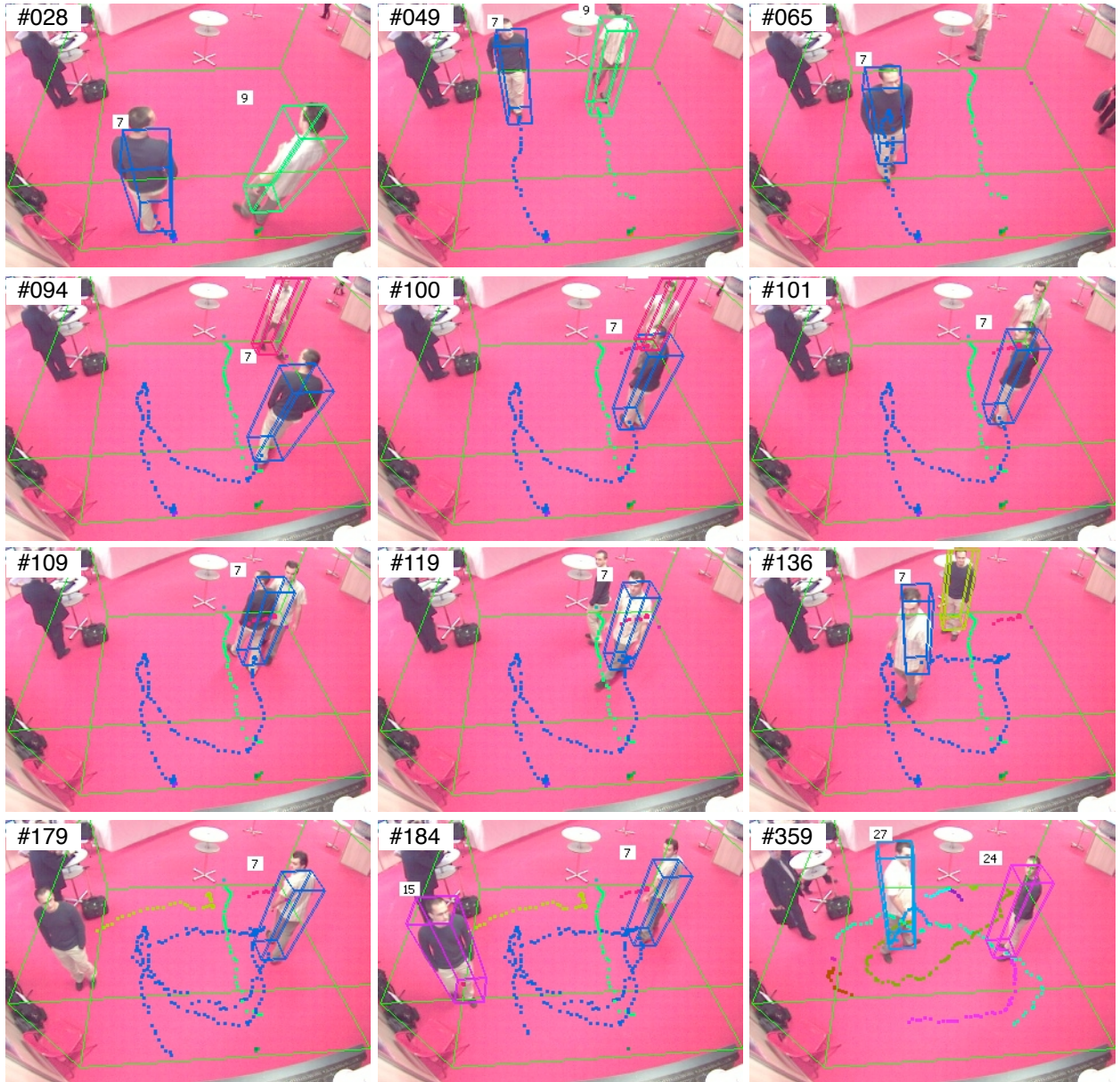


FIGURE 5.6 – Extraits de la séquence de suivi de piétons sur un espace très réduit ( $3.5 \times 3.5$  mètres) mettant en évidence les pertes d'identité lorsqu'une cible sort de l'espace de traque (ou s'approche du bord).



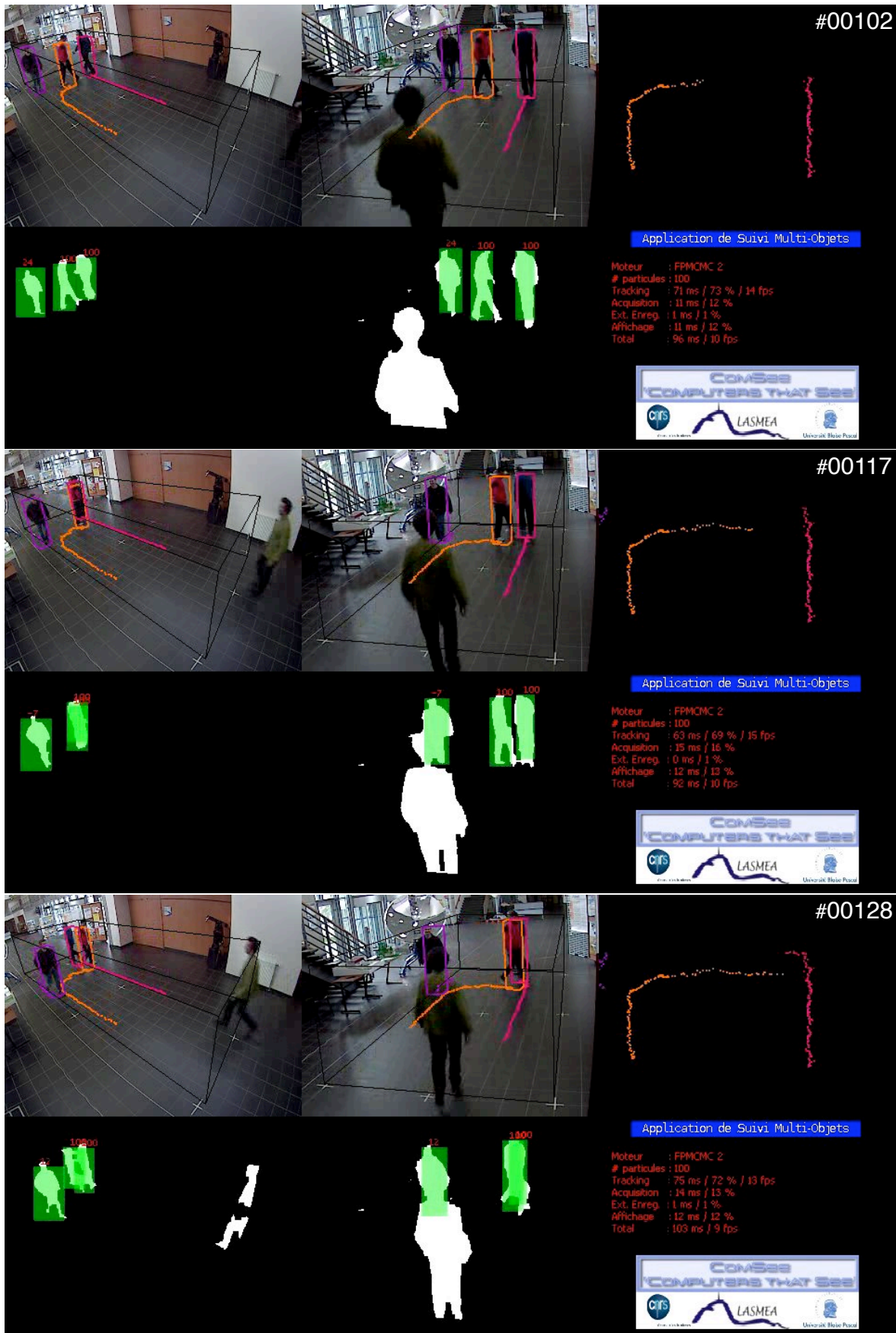


FIGURE 5.7 – Suivi de piétons avec 2 caméras. Gauche : caméra 1, milieu : caméra 2, droite : trajectoires en vue de dessus. Pour chaque instant : Haut : images brutes et piétons estimés. Bas : segmentation *fond/forme*, rectangles englobants, et *vitalité*  $\times 100$ .

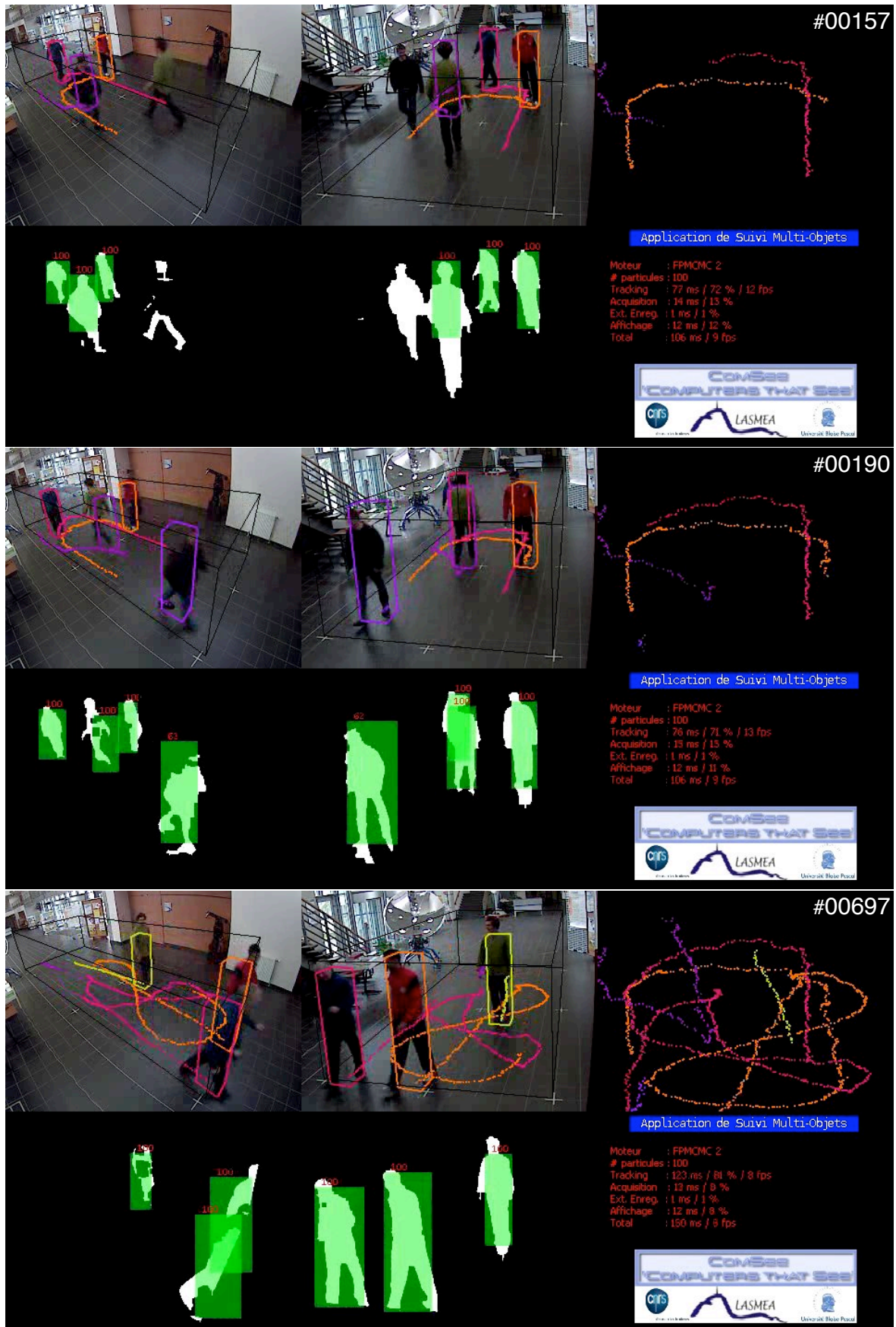


FIGURE 5.8 – Suivi de piétons avec 2 caméras. Gauche : caméra 1, milieu : caméra 2, droite : trajectoires en vue de dessus. Pour chaque instant : Haut : images brutes et piétons estimés. Bas : segmentation *fond/forme*, rectangles englobants, et *vitalité*  $\times 100$ .

TABLE 5.2 – Erreur moyenne de position par rapport à la mesure GPS

séquence	moyenne (m)	écart-type (m)
entrée de courbe	0.99	0.067
sortie de courbe	0.15	0.039

mesure de la précision du suivi, en référence à une vérité délivrée par un GPS centimétrique fixé sur la moto. La figure 5.9 page 115 donne une vue d'ensemble de l'entrée et de la sortie de la courbe où se fait la mesure de trajectoire. L'acquisition vidéo et la vérité terrain par GPS centimétrique ont été réalisées par le LCPC<sup>3</sup>. Quatre séquences ont été tournées :

- ▷ entrée de courbe en suivant le bord droit de la chaussée,
- ▷ entrée de courbe en suivant la ligne médiane de la chaussée (illustrée par la vue en haut à gauche de la figure 5.9 page 115,
- ▷ sortie de courbe en suivant le bord droit de la chaussée,
- ▷ sortie de courbe en suivant la ligne médiane de la chaussée (illustrée par la vue en haut à droite de la figure 5.9 page 115.

Le suivi a été répété 10 fois sur chacune des 4 séquences. La moyenne et la variance de l'erreur d'estimation de trajectoire par rapport au GPS sont indiquées dans le tableau 5.2. L'erreur est beaucoup plus élevée pour les séquences d'entrée courbe que pour les séquences de sortie de courbe. Les causes sont :

- ▷ l'initialisation est beaucoup plus précise pour les objets proches de la caméra,
- ▷ l'ombre de la moto biaise nettement plus la position sur les séquences d'entrée, du fait de la position du soleil.

Nos résultats se rapprochent des besoins de précision des analyses de trajectoires sur les séquences de sortie de courbe. Mais l'erreur des séquences d'entrée est beaucoup trop élevée, car l'ombre portée au sol de la moto est classifiée comme avant-plan, ce qui biaise considérablement le suivi. Les images de gauche de la figure 5.9 page 115 montrent clairement le biais de position de l'objet estimé (cube orange sur l'image du haut) causé par ce défaut de segmentation *fond / forme* (en bas). Ce constat nous mène au besoin de modéliser l'ombre des objets portée sur le sol, qui fait l'objet de la campagne d'essais dont les résultats sont rapportés en section 5.5.

## 5.5 Suivi de piétons sous illumination variable

Les données vidéos sont extraites de campagnes d'acquisition typiques de vidéo-surveillance de lieux publics. Les piétons candidats décrits par les particules, sont contrôlés par des propositions de vitesse tirées de la loi (4.5). La forme est mise à jour pour s'adapter à l'objet cible, selon l'équation (4.6). Le système est testé dans deux versions, avec et sans modélisation des ombres des piétons supposées portées sur le sol :

- ▷ *MOT* - Suivi Multi Objet : implémentation de l'algorithme *FP RJ-MCMC* sans classification ni modélisation de l'ombre. Une seule catégorie d'objets candidats est alors utilisée. La dynamique de taille d'objet est alors fortement augmentée, pour permettre aux objets candidats de s'adapter à des cibles de tailles aussi différentes que poids lourds et motos.
- ▷ *MOTS* - Suivi Multi Objet et modélisation de l'ombre : implémentation de l'algorithme *FP RJ-MCMC* sans classification. L'éclairement est modélisé par une source ponctuelle située à l'infini, et l'ombre est

<sup>3</sup>[www.lcpc.fr](http://www.lcpc.fr)



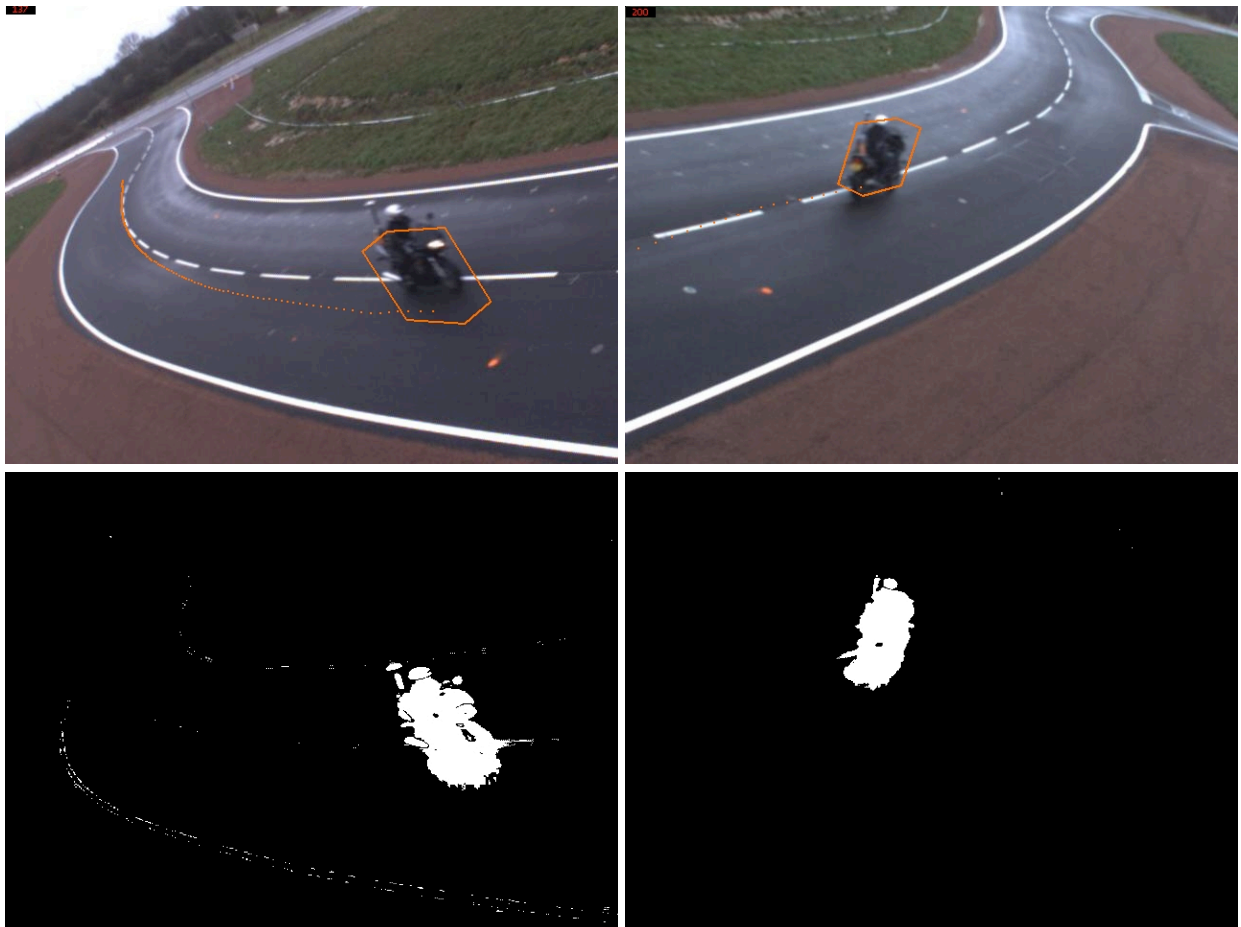


FIGURE 5.9 – Gauche : entrée de courbe. Droite : sortie de courbe. Enveloppe estimée et trajectoire de la moto. Haut : images couleur. Bas : images binaires d'avant-plan.

TABLE 5.3 – Suivi de parallélépipèdes approximant des piétons sur séquences de synthèses. Dans la colonne de gauche, le soleil est direct (sans nuage), mais la direction de l'éclairement varie avec le temps. Dans la colonne de droite, une alternance d'éclaircies et de nuages est générée toutes les 200 images, et la direction de l'éclairement varie avec le temps.

	soleil direct		soleil & nuages	
	<i>MOT</i>	<i>MOTS</i>	<i>MOT</i>	<i>MOTS</i>
$\theta_T$ (%)	84.7	89.7	84.1	87.0
$\theta_G$ (%)	5.7	4.9	5.1	4.3
erreur (m)	0.91	0.63	0.82	0.70

supposée portée soit sur un sol plan, soit sur les autres objets. Une seule catégorie d'objets candidats est utilisée. Dynamique de taille d'objet élevée pour permettre aux objets candidats de s'adapter à des cibles de tailles aussi différentes que poids lourds et motos.

**Séquences de synthèse :** Des parallélépipèdes approximant des piétons se déplacent de façon aléatoire sur une zone de suivi de  $12 \times 15$  mètres. L'évolution de l'illumination est simulée et fortement accélérée, afin de valider la capacité du système à s'adapter à des variations rapides d'illumination : altitude  $\psi = 0.8 \text{ rad}$ , azimut variant de  $\phi = 0$  à  $\phi = \pi \text{ rad}$  en 1000 images. La figure 5.10 page 117 illustre le suivi, et les résultats quantitatifs sont portés dans le tableau 5.3, montrant l'intérêt de la modélisation de l'ombre : les trois critères  $\theta_T$ ,  $\theta_G$ , et l'erreur de position sont améliorés dans les deux cas d'ensoleillement.

**Séquence réelle :** Une courte séquence captée lors d'une alternance de nuages et d'éclaircies, provoquant de rapides variations d'illumination. L'image #786 de la figure 5.11 page 118 illustre le suivi de trois piétons, lorsque le temps est estimé couvert, le système ne modélisant alors pas d'ombres portées. Quelques images plus tard, comme l'ensoleillement devient clair le traqueur estime que le soleil apparaît à l'image #823, et reste clair jusqu'à la fin. Le traqueur échoue à estimer les deux piétons cibles marchant côte à côte car ils s'occultent mutuellement sur l'ensemble de la séquence : il les suit comme un seul piéton (#14), en raison de l'absence d'observabilité.

## 5.6 Suivi et catégorisation de véhicules sous illumination choisie.

On verra que les ombres portées des objets perturbent le suivi. Il faudra donc les modéliser. Auparavant, nous avons voulu évaluer l'effet du poids d'interaction entre objets présentée dans la section 4.8. Les séquences ont été choisies sous illumination diffuse, afin d'éviter les effets des ombres trop marquées. Le suivi et catégorisation simultanés de véhicules dans le trafic implique des voitures, camionnettes, et poids lourds. Les véhicules sont contrôlés par des propositions de commandes de conducteur tirées de la loi (4.58). Les équations du modèle bicycle (4.56) et (4.57) sont ensuite appliqués à l'objet  $j$ . Les tableaux 5.4 et 5.5 exposent les résultats, respectivement avec un potentiel d'interaction isotrope, et anisotrope, telle que décrite en section 4.8. La figure 5.12 page 119 permet de comparer les deux, et met en évidence l'intérêt de cette dernière, notamment pour diminuer le taux d'objets candidats fantômes  $\theta_G$ . Elle montre cependant que le taux de classification des camionnettes (la catégorie de taille intermédiaire) est très faible. Une analyse plus poussée des résultats montre que 100 % de camionnettes mal classées sont suivies comme étant des véhicules légers. Les dimensions de ces modèles ne sont pas très éloignées, comme on peut le voir dans le tableau 4.1. Ce résultat montre les limites de la classification basée sur un modèle géométrique aussi grossier que celui que nous avons retenu.

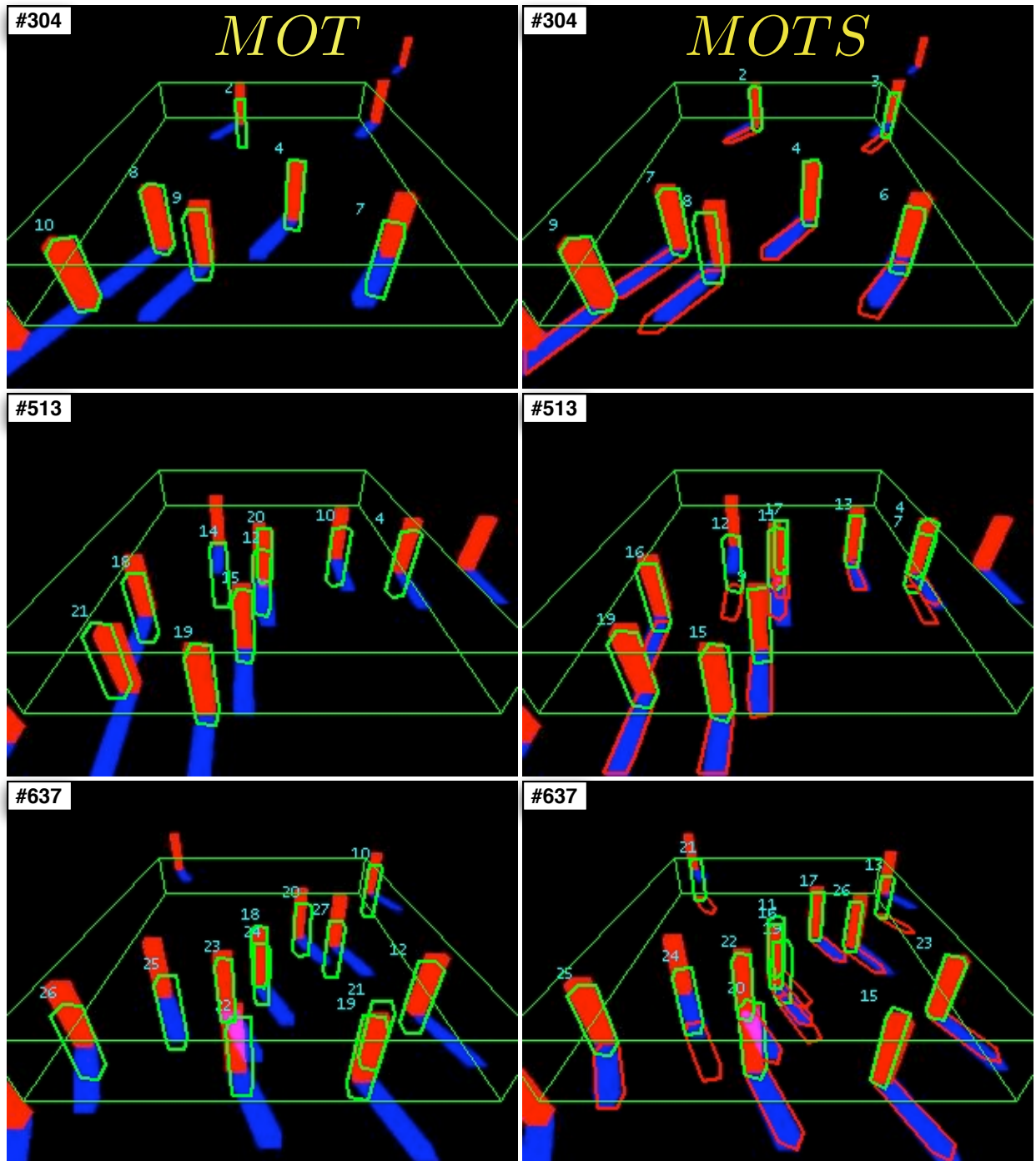


FIGURE 5.10 – Extraits du suivi de piétons sur vidéos de synthèse, avec variation rapide d’azimut d’un soleil simulé. La projection des parallélépipèdes estimés est tracée en vert. Colonne de gauche : pas de modélisation de l’ombre. Colonne de droite : avec modélisation de l’ombre, l’estimation des ombres portées au sol étant tracée en rouge.



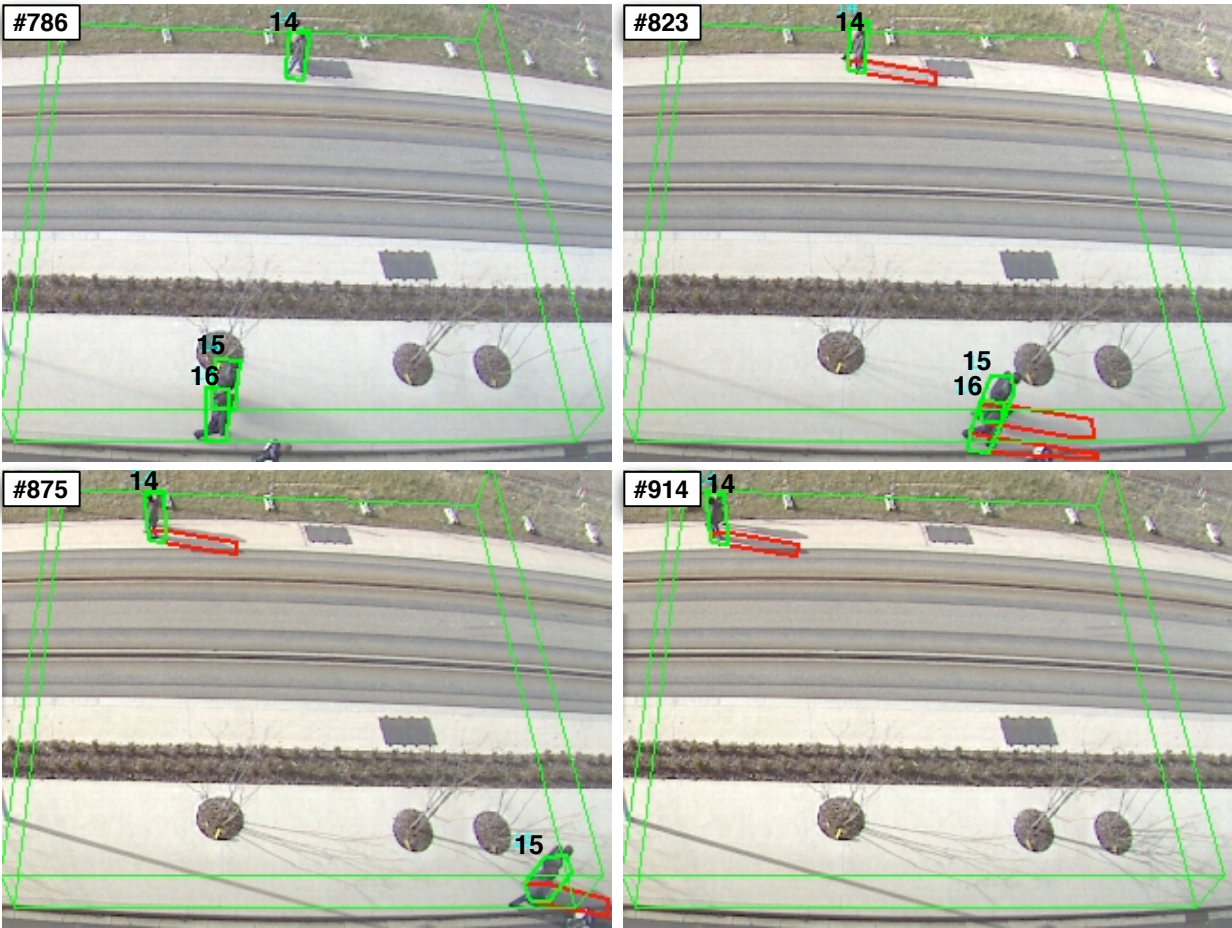


FIGURE 5.11 – Extraits du suivi de piétons sur vidéos réelles, ensoleillement variable (nuages et éclaircies). Estimation des contours d’objets en trait vert, estimation des contours d’ombres portées en traits rouges lorsque le soleil est estimé comme brillant.

TABLE 5.4 – Suivi et Classification de véhicules en 3 classes. Potentiel d’interaction isotrope.

	<b>véhicules légers</b>	<b>camionnettes</b>	<b>semi-remorques</b>	<b>total</b>
Nombre de véhicules	330	58	51	439
$\theta_T$ (%)	97	100	100	98
$\theta_C$ (%)	97	57	92	91
$\theta_G$ (%)	9	34	8	12

TABLE 5.5 – Suivi et Classification de véhicules en 3 classes. Potentiel d’interaction anisotrope.

	<b>véhicules légers</b>	<b>camionnettes</b>	<b>semi-remorques</b>	<b>total</b>
Nombre de véhicules	330	58	51	439
$\theta_T$ (%)	96	100	100	97
$\theta_C$ (%)	95	52	100	90
$\theta_G$ (%)	8	3.5	0	6

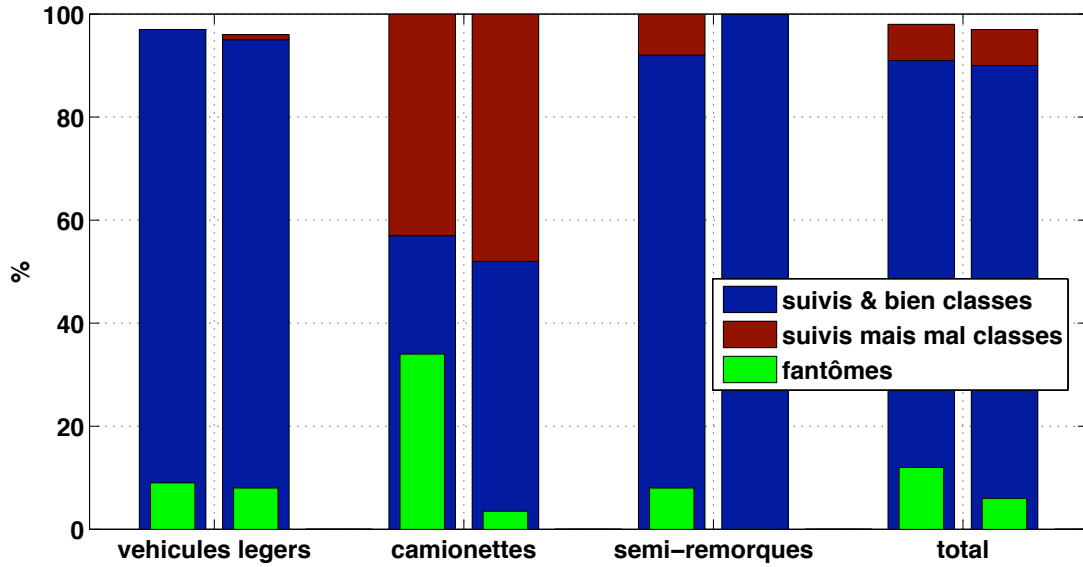


FIGURE 5.12 – Taux de suivi  $\theta_T$ , taux de classification  $\theta_C$ , et taux de fantômes  $\theta_G$  pour 3 classes de véhicules. Pour chacune, les barres de gauche sont pour un potentiel d’interaction isotrope, et les barres de droite sont pour un potentiel d’interaction anisotrope.

## 5.7 Suivi et catégorisation de véhicules sous illumination variable

Les expérimentations présentées dans cette section ont été menées sous diverses conditions d’illuminations, notamment avec soleil et nuages, à plusieurs heures du jour. Cette campagne d’expérimentations a pour but de quantifier la performance du système sur du suivi et catégorisation simultanés de véhicules dans le trafic, impliquant des voitures, camionnettes, poids lourds. Le contrôle des véhicules candidats reste conforme au modèle exposé dans la section 5.6. Les critères  $\theta_T$ ,  $\theta_C$ ,  $\theta_G$ ,  $\epsilon_T$  ont été utilisés pour comparer les performances de quatre versions du système :

- ▷ *MOT* - Suivi Multi Objet : implémentation de l’algorithme *FP RJ-MCMC* sans classification ni modélisation de l’ombre. Une seule catégorie d’objets candidats est alors utilisée. La dynamique de taille d’objet est alors fortement augmentée, pour permettre aux objets candidats de s’adapter à des cibles de tailles aussi différentes que poids lourds et motos.
- ▷ *MOTS* - Suivi Multi Objet et modélisation de l’ombre : implémentation de l’algorithme *FP RJ-MCMC* sans classification. L’éclairement est modélisé par une source ponctuelle située à l’infini, et l’ombre est supposée portée soit sur un sol plan, soit sur les autres objets. Une seule catégorie d’objets candidats est utilisée. Dynamique de taille d’objet élevée pour permettre aux objets candidats de s’adapter à des cibles de tailles aussi différentes que poids lourds et motos.
- ▷ *MOTC<sup>n</sup>* - Suivi et Catégorisation Multi Objet sans modélisation de l’ombre : implémentation de l’algorithme *FP RJ-MCMC* avec catégorisation des objets en  $n$  classes.
- ▷ *MOTC<sup>n</sup>S* - Suivi et Catégorisation Multi Objet avec Modélisation de l’ombre : implémentation de l’algorithme *FP RJ-MCMC* avec catégorisation des objets en  $n$  classes. L’éclairement est modélisé par une source ponctuelle située à l’infini, et l’ombre est supposée portée soit sur un sol plan, soit sur les autres objets.

**Séquences vidéos de synthèse :** Les vidéos ont été réalisées avec une caméra fictive placée au bord d’une ligne droite d’autoroute à trois voies. Des parallélépipèdes approximant deux catégories de véhicules« cir-

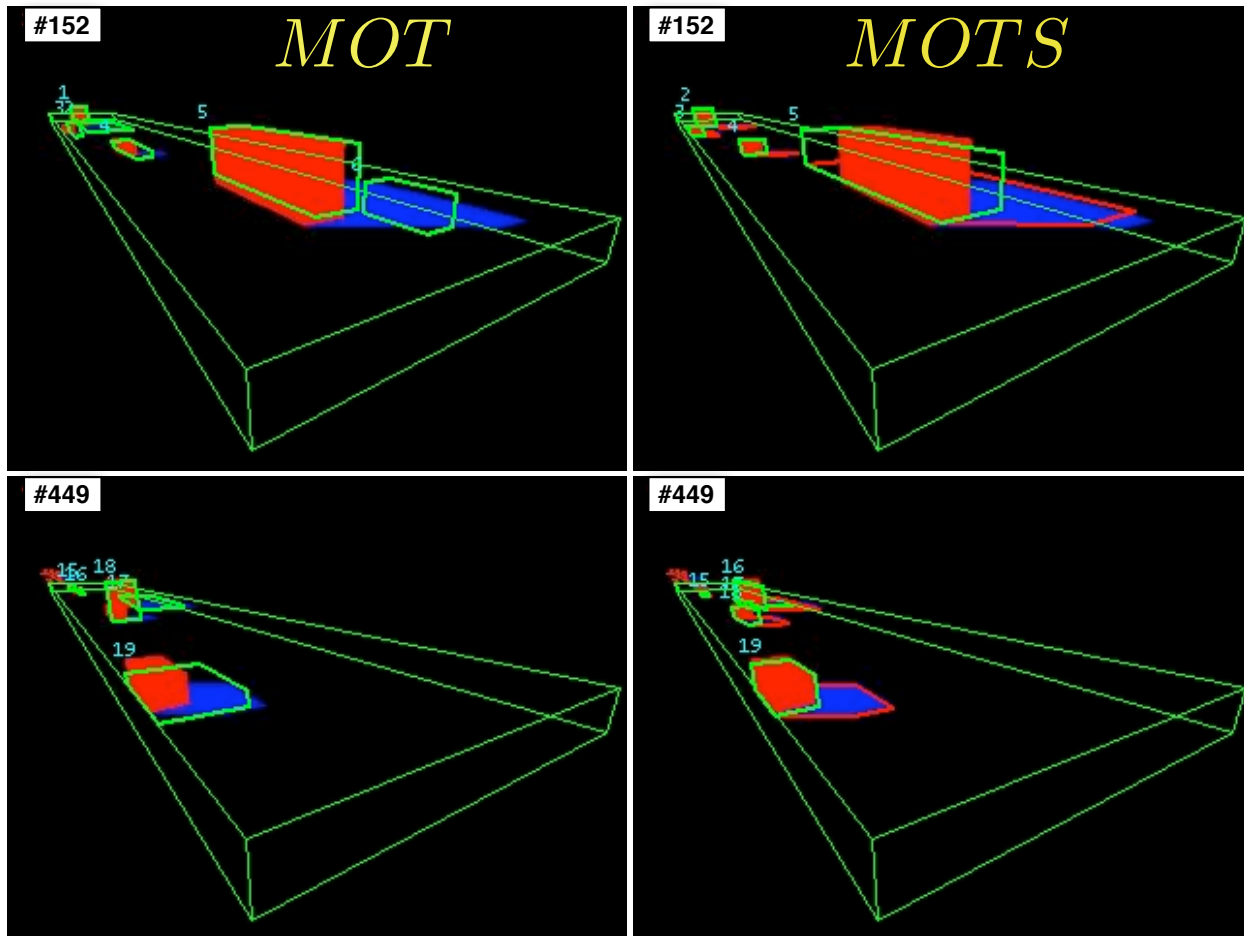


FIGURE 5.13 – Extraits d’une des séquences de synthèse, avec suivi de cubes approximant deux catégories de véhicules sur autoroute : voitures et poids lourds. Aucune catégorisation n’est menée dans cette figure. Images de gauche : sans modélisation de l’ombre. Estimation des contours d’objets en trait vert. Images de droite : avec modélisation de l’ombre, estimation des contours d’ombres en trait rouge.

culent » sur cette autoroute, sous un soleil brillant. La figure 5.13, colonne de gauche (*MOT*), illustre deux dysfonctionnement du suivi sans modélisation de l’ombre. Apparition d’objets fantômes sur les ombres : à l’image #152, l’objet estimé #6 ne suit que l’ombre de l’objet #5. Estimation de forme et de position des objets fortement biaisées par leurs ombres : à l’image #449, l’objet estimé #19 est attiré par l’ombre de la cible, causant un biais de position pouvant atteindre 2 mètres. Ces deux dysfonctionnements sont réglés par la modélisation de l’ombre, colonne de droite, (*MOTS*). Cependant, la modélisation de l’ombre ne suffit pas à régler tous les problèmes : la figure 5.14 illustre qu’en absence de catégorisation, le système peut expliquer un unique objet long par plusieurs objets courts. Le scénario illustré par cette figure se produit fréquemment et est dû à l’entrée progressive dans le champ de mesure d’un objet cible de grandes dimensions. Dans l’image #385 (en haut), le système initialise un objet candidat (#16) de petites dimensions sur cette cible encore partiellement invisible. A l’image #387, la visibilité accrue de l’objet permet au système *MOTC<sup>2</sup>S* (à droite) de changer la catégorie de l’objet candidat. Le système sans catégorisation *MOTS* initialise à l’image #392 un deuxième objet candidat (#17) de petites dimensions suivant le premier. Comme illustré par l’image #398, il sera difficile de récupérer un suivi correct dans ce cas, tandis que *MOTC<sup>2</sup>S* a déjà bien placé l’objet candidat sur la cible. Le tableau 5.6 rapporte les résultats, montrant que la modélisation de l’ombre et l’activation de la catégorisation d’objets améliorent toutes deux indépendamment les performances. Les meilleurs résultats sont obtenus lorsque les deux sont activées.

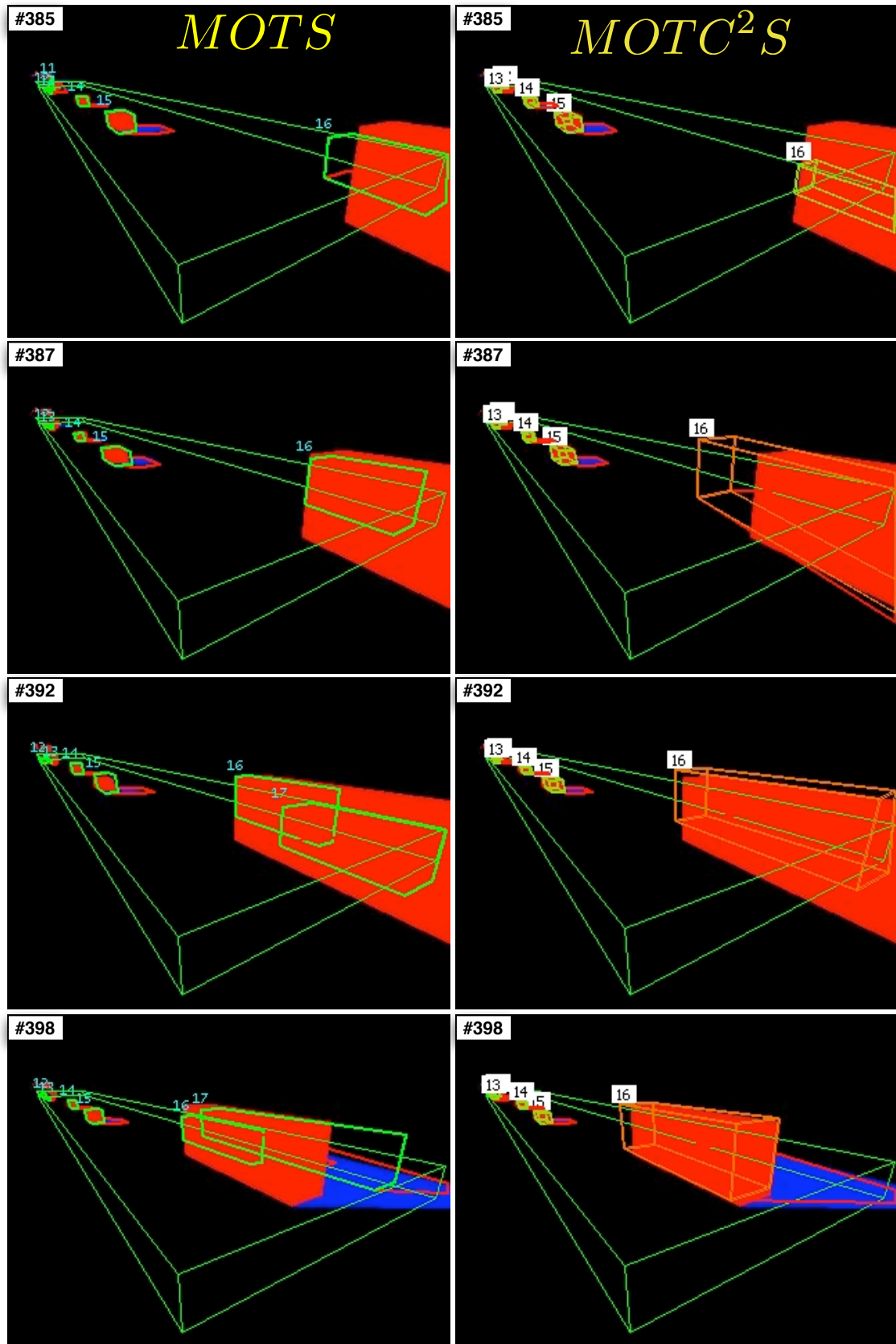


FIGURE 5.14 – Extraits d’une des séquences de synthèse, avec suivi de cubes approximant deux catégories de véhicules sur autoroute : voitures et poids lourds, avec modélisation de l’ombre. Images de gauche : sans catégorisation. Images de droite : avec catégorisation en 2 classes : voitures en vert, poids lourds en orange.

TABLE 5.6 – Vidéos de synthèse sur autoroute, avec véhicules approximatés par des cubes de deux catégories. Suivi et catégorisation. Taux de suivi  $\theta_T$  (%) / Taux de catégorisation  $\theta_C$  (%) / Taux de fantômes  $\theta_G$ (%). erreur moyenne de position par vehicule (mètres).

	<i>MOT</i>	<i>MOTS</i>	<i>MOTC</i> <sup>2</sup>	<i>MOTC</i> <sup>2</sup> S
véhicules légers	.	.	59/54/0	<b>90/89/11</b>
poids lourds	.	.	86/86/0	<b>90/89/0</b>
<b>total</b>	52/22/17	51/25/16	58/53/0	<b>90/89/11</b>
<b>erreur (m)</b>	6.17	5.80	2.76	<b>2.00</b>

TABLE 5.7 – Suivi et catégorisation de véhicules réels sur autoroute. Taux de suivi  $\theta_T$  (%) / Taux de catégorisation  $\theta_C$  (%) / taux de fantômes  $\theta_G$ (%). erreur moyenne de position par véhicule (mètres).

	<i>MOT</i>	<i>MOTS</i>	<i>MOTC</i> <sup>3</sup>	<i>MOTC</i> <sup>3</sup> S
véhicules légers	.	.	67/64/2.6	<b>67/67/0.05</b>
camionnettes	.	.	83/36/1.0	<b>92/86/3.7</b>
poids lourds	.	.	93/83/0	<b>100/100/2</b>
<b>total</b>	51/45/0	60/51/0	72/62/2.5	<b>70/70/3.1</b>
<b>erreur (m)</b>	6.80	6.22	6.13	<b>5.40</b>

**Séquences de trafic réel :** Les séquences de trafic réel utilisées impliquent des véhicules légers, des camionnettes et des poids lourds, sur une autoroute à quatre voies. La scène est captée à la sortie d'une barrière de péage, et comporte une voie d'accélération. Les conditions d'illuminations sont variables. Pour faire du suivi de véhicules en trafic réel, la classification est indispensable. En effet, du fait de la grande étendue de taille des véhicules, les méthodes sans classification (*MOT* et *MOTS*) requièrent une valeur élevée de la dynamique de forme des véhicules  $\sigma_s$ , afin de permettre aux objets candidats de s'adapter à la grande variété des objets cibles. Il faut donc prendre en compte les catégories les plus fréquentes. Une telle stratégie ne peut pas fonctionner en présence de fortes occultations. Pour qu'elle puisse constituer une référence, ainsi que pour nous permettre d'établir la vérité terrain à la main, nous avons choisi des séquences de trafic modéré, mais impliquant toutes les catégories de véhicules. La figure 5.15 illustre que la catégorisation et la modélisation de l'ombre améliorent le suivi. Les défaillances typiques de *MOT* et *MOTS* sont : deux objets candidats suivant une cible unique (*MOTS*) ou mauvaise précision du suivi (*MOT*). Sans modélisation de l'ombre, le système ne parvient pas à suivre des objets de tailles très différentes : il explique les pixels classés comme avant-plan appartenant à l'ombre du camion, par un « fantôme » de voiture (#7 sur *MOTC*<sup>3</sup> et #8 sur *MOTC*<sup>3</sup>S). La modélisation de l'ombre portée permet d'expliquer correctement ces pixels d'avant plan (*MOTC*<sup>3</sup>S). En outre, les véhicules sont plus précisément localisés lorsque l'ombre est modélisée (*MOTS* et *MOTC*<sup>3</sup>S), car l'utilisation de l'ombre accroît l'observabilité de l'objet, notamment sur leur position longitudinale. Le tableau 5.7 regroupe les résultats et confirme les analyses de la section 5.7 : la modélisation de l'ombre et la catégorisation d'objets améliorent toutes deux indépendamment les performances. Les meilleurs résultats sont obtenus lorsque les deux sont activées.

## 5.8 Suivi et catégorisation de véhicules sur différentes situations

Cette campagne d'essais a pour objectif de montrer qualitativement que les modèles dynamiques de véhicules utilisés, sont suffisamment peu contraints pour permettre à l'algorithme de traiter tout type de séquence routière, sans aucune adaptation *ad-hoc* des modèles dynamiques des véhicules. Une première expérimentation est menée sur l'arrivée à une barrière de péage, une deuxième expérimentation est menée sur une section de route sinueuse, enfin des expérimentations sont en cours sur des carrefours giratoires.



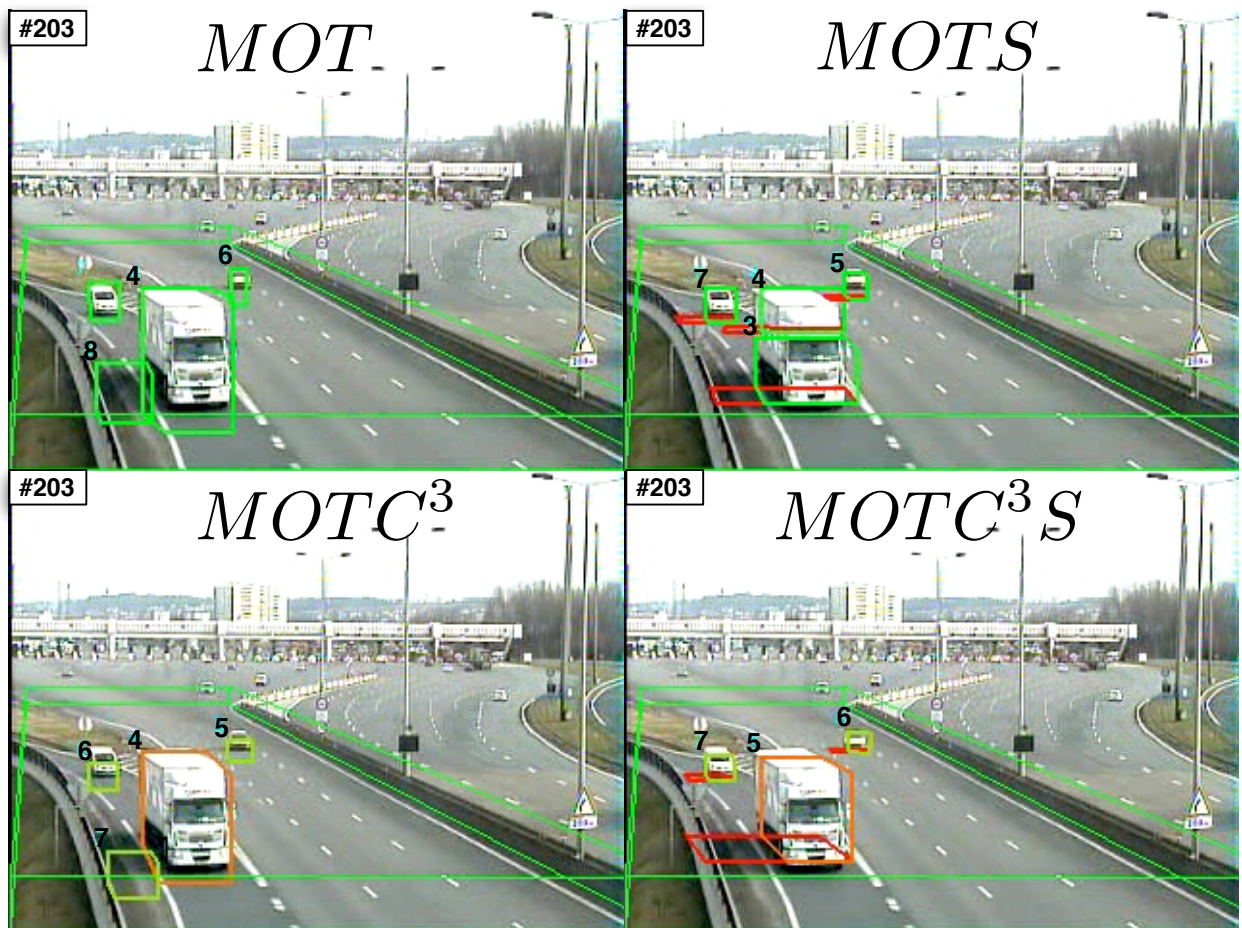


FIGURE 5.15 – Une image extraite (#203) d’une séquence de suivi de véhicules en trafic réel. Haut : pas de classification, estimation des contours d’objets en trait vert. Bas : classification en 3 catégories. Estimation des contours des véhicules légers en trait vert, ceux des camionnettes en magenta, et ceux des poids lourds en orange. A gauche : sans modélisation des ombres. A droite : avec modélisation des ombres, estimation des contours d’ombres portées en traits rouges lorsque le soleil est estimé comme brillant.

### 5.8.1 Suivi et catégorisation de véhicules sur l'entrée d'une barrière de péage

Cette expérimentation se caractérise par une décélération longitudinale des véhicules relativement élevée (de l'ordre de  $2m.s^{-2}$ ), un fort effet de perspective, et une très mauvaise observabilité de la vitesse et de la position longitudinale des véhicules entrant de la zone de suivi. La scène est illustrée sur la figure 5.16 page 125. Une difficulté supplémentaire est due aux reflets des véhicules sur la chaussée mouillée. Ces reflets biaisent l'estimation de position des véhicules, comme on peut le constater sur les images. Sur cette séquence, 100% des véhicules ont été correctement suivis et classés. Les résultats sont corrects mais ont été obtenus grâce à une initialisation supervisée de la vitesse des véhicules.

### 5.8.2 Suivi et catégorisation de véhicules sur route sinueuse

Cette expérimentation se caractérise par une décélération transversale des véhicules modérée (de l'ordre de  $2m.s^{-2}$ ) et un fort effet de perspective. Le suivi des véhicules a été menée sur une séquence de 5.500 images ( $480 \times 640$  enregistré à 25 images par seconde), impliquant 37 véhicules : 20 véhicules légers, 15 motos et 2 camionnettes. La scène est illustrée sur la figure 5.17 page 126, les lignes noires délimitant la zone de suivi de  $160 \times 6$  mètres. Sur cette séquence, 100% des véhicules ont été correctement suivis et classés.

### 5.8.3 Suivi et catégorisation de véhicules sur carrefour giratoire

Les expérimentations sont en cours, sur un carrefour giratoire tel qu'illustré par la figure 5.18 page 127. Le besoin est la connaissance de la matrice de flux d'entrées-sorties de véhicules sur le giratoire, classiquement menée manuellement. La caméra est placée sur une nacelle mobile, à 12 mètres d'altitude. Nous n'avons pas encore de résultats de suivi et classification, car il faut modifier la procédure d'initialisation des vitesses des véhicules, qui peuvent entrer dans la zone de suivi en venant de toutes les directions et à des vitesses très variables. D'autres difficultés doivent être surmontées afin que cette application soit opérationnelle :

- ▷ L'élasticité du bras de la nacelle la rend très sensible au vent, ce qui impose une stabilisation de l'image.
- ▷ La forte présence de végétation également très sensible au vent peut perturber l'observation.
- ▷ Présence d'objets occultants sur le terre-plein.
- ▷ Très forte occultation mutuelle des objets en cas de file d'attente, comme cela se produit en haut à gauche de l'image #00156 de la figure 5.18, où 6 voitures s'occultent mutuellement.

## 5.9 Conclusion

Dans une première partie de ce chapitre, nous avons comparé la capacité de trois configurations du *FP RJ-MCMC* à maintenir l'identité des objets à travers des périodes d'invisibilité de ces objets. La résistance à des périodes d'invisibilité forte ou totale est rendue nécessaire du fait des occultations pouvant se produire, causées soit par un autre objet, soit par un élément occultant du fond, derrière lequel l'objet passe, tel qu'un arbre, un panneau, un pont... Les résultats montrent une très nette supériorité de la cohérence temporelle des identités lorsque la *vitalité*, introduite au chapitre 4, est activée. Les expérimentations sur vidéo réelles sont moins favorables à nos choix, et montrent qu'avec les contraintes imposées par nos applications, la cohérence temporelle des identités reste faible en l'absence de descripteurs d'apparence. Dans une deuxième partie, nous avons décrit et analysé une campagne d'expérimentations visant à quantifier les performances de notre système de suivi et classification conjoints d'un nombre variable d'objets sous illumination variable, dans une grande variété de situations. Les expérimentations ont été menées sur piétons, motos, véhicules légers, camionnettes et poids lourds. Des critères ont été proposés pour quantifier les performances de suivi et de classification des objets. Les résultats montrent que la modélisation de l'ombre et la catégorisation d'objets améliorent toutes deux indépendamment les performances. Les meilleurs résultats sont obtenus lorsque les deux sont activées.



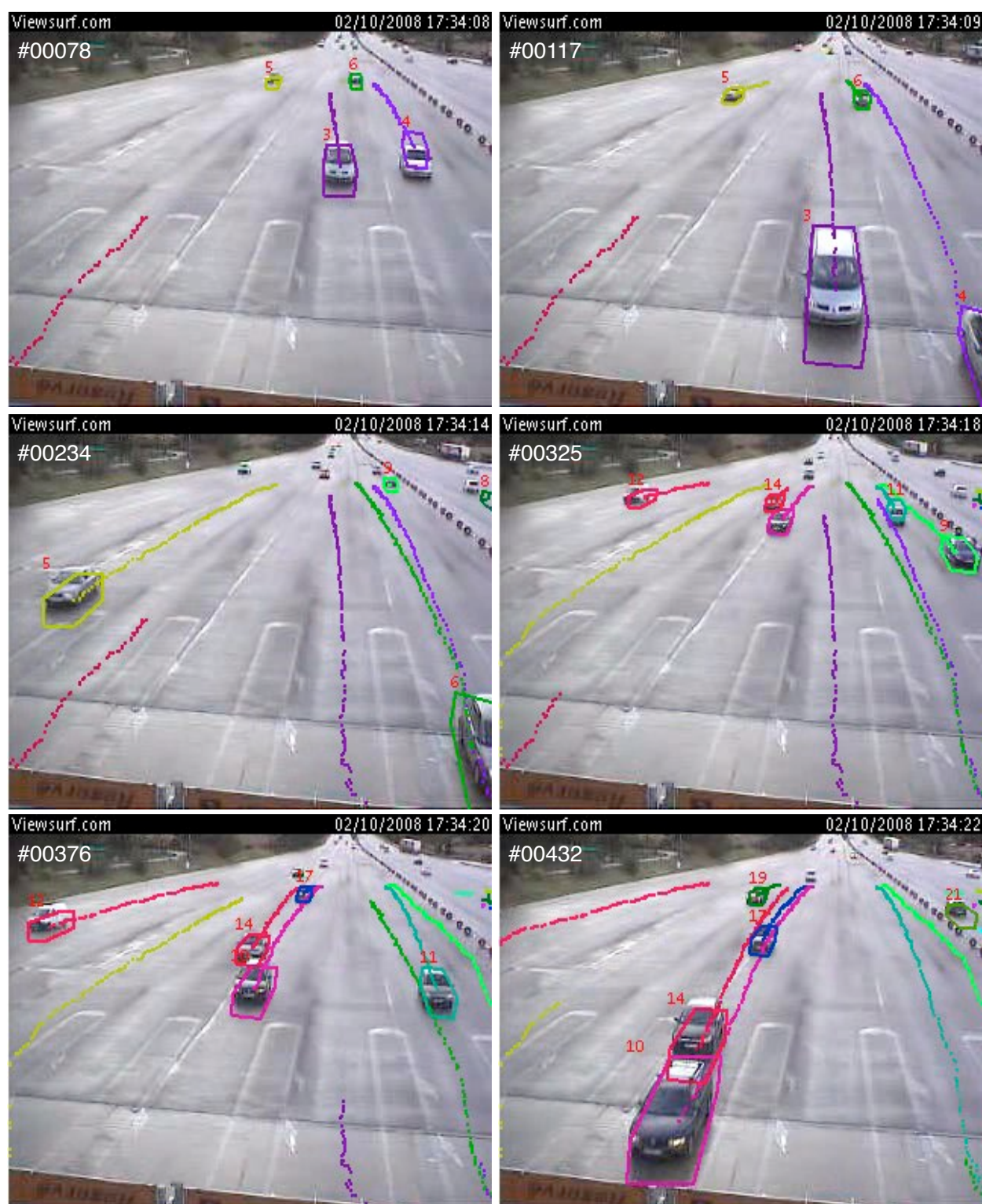


FIGURE 5.16 – Enveloppe estimée et trajectoires de véhicules sur autoroute à l'arrivée à une barrière de péage. La zone de suivi est un rectangle de  $140 \times 24$  mètres.



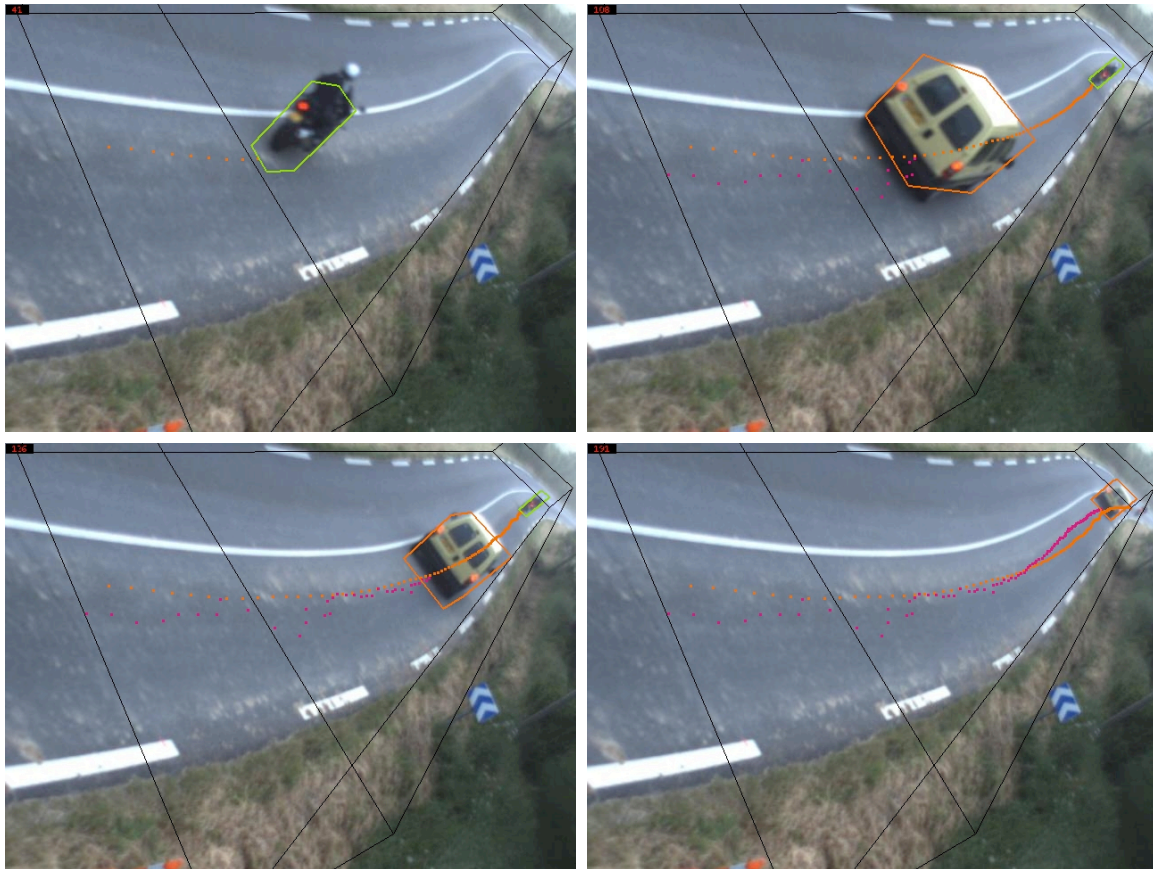


FIGURE 5.17 – Enveloppe estimée et trajectoires motos et voitures sur route sinueuse (la zone de suivi est délimitée par les lignes noires).



FIGURE 5.18 – Extraits d’une séquence de mesure des flux d’entrée-sortie sur carrefour giratoire.



---

## CONCLUSION ET PERSPECTIVES

Nous avons présenté dans ce mémoire un traqueur multi-objets générique pouvant être utilisé dans de multiples applications. Nous avons visé en particulier les applications nécessitant le suivi en temps réel d'une scène dynamique complexe, impliquant un nombre élevé d'objets, nécessité par la surveillance de lieux publics et d'infrastructures routières. Le cahier des charges lié à ces applications et les choix que nous avons faits ont été spécifiés au chapitre 1. Rappelons les brièvement : 1) capteurs optiques, 2) traitement causal des observations pour le suivi en temps réel, 3) suivi en 3D permettant de modéliser la dynamique des objets, la projection perspective, et les occultations, 4) modèle d'observation basé sur une segmentation *fond / forme* de l'image courante, et prenant en compte la distance de chaque objet à chaque caméra, 5) méthode d'inférence probabiliste Bayésienne, 6) représentation non paramétrique des lois de probabilité, 7) suivi de la configuration jointe multi-objets par un unique filtre opérant dans l'espace d'état joint. Ce dernier choix est justifié par la fréquence des interactions et occultations mutuelles entre objets dans les applications visées. Nous avons montré au chapitre 1 que le choix d'un filtre unique opérant sur l'espace joint de la configuration globale multi-objets permet de modéliser ces interactions, ce qui améliore nettement les performances du suivi. Ce choix entraîne néanmoins une difficulté majeure, puisqu'il impose au filtre d'opérer dans un espace de grande dimension. Comme nous avons choisi par ailleurs une représentation non paramétrique donc échantillonnée des lois de probabilité, nous nous trouvons face au délicat problème de l'échantillonnage de ces lois dans un espace de grande dimension, par un nombre modeste d'échantillons, en vue du temps réel. Plusieurs stratégies d'échantillonnage existantes ont été présentées au chapitre 2, et la comparaison de leurs performances menée au chapitre 3 donne l'avantage au *Filtre Particulaire MCMC* à propositions marginales. Nous avons également proposé dans le chapitre 2 une extension de ce filtre permettant d'exploiter les capacités de calculs parallèles des machines actuelles, et les tests de performances menés au chapitre 3 montrent l'intérêt de cette extension. La réalisation, détaillée au chapitre 4, s'appuie notamment sur la méthode de suivi d'un nombre variable d'objet par *RJ MCMC*, développée par Khan *et al.* (KBD05). Par rapport à cette référence, nos contributions sont présentées dans le chapitre 4, et brièvement rappelées ci-dessous.

### Contributions :

1. Fonction de vraisemblance relative à la distance objet-caméra, qui améliore l'efficacité de l'algorithme de Metropolis-Hastings, en asservissant son taux d'acceptation.
2. Filtre à propositions multiples  $FP\ MCMC^P$ , approximation du  $FP\ MCMC^1$  pour améliorer les performances temps réel en exploitant les capacités de calculs parallèles des machines actuelles.
3. *Vitalité* des objets pour contrôler leur temps de survie à une mauvaise observation, due à une occultation ou à une mauvaise segmentation *fond / forme*.
4. Suivi et catégorisation assurés conjointement par le filtre. La catégorie de l'objet est intégrée dans l'état de chaque objet et suivie par le filtre au même titre que les autres variables.
5. Modélisation et suivi de l'illumination également pris en charge par le filtre, pour minimiser les perturbations causées par les ombres portées des objets.

**Résultats :** Nous avons mené une première campagne d'expérimentation pour évaluer l'efficacité de la *vitalité* en vue de maintenir l'identité des objets à travers des périodes d'invisibilité de ces objets. Les résultats montrent son intérêt. Le filtre, très générique, est capable de suivre des objets aussi différents que des piétons, motos, véhicules légers, camionnettes et poids lourds. Les expérimentations présentées dans le chapitre 5 montrent son efficacité, dans le respect des contraintes spécifiées par le cahier des charges. Elles montrent également que la modélisation de l'ombre et la catégorisation d'objets améliorent toutes deux indépendamment les performances, et sont tout à fait compatibles. On a donc intérêt à les mettre en œuvre toutes deux pour obtenir les meilleurs résultats.

**Perspectives : amélioration des performances.** Le système présenté dans ce mémoire reste une méthode générique, dont les performances peuvent être grandement améliorées par les travaux suivants :

- ▷ **Modélisation de l'apparence individuelle des objets.** La fonction de vraisemblance utilisée est incapable de distinguer l'identité des objets. Si nos résultats présentent une bonne cohérence temporelle correcte, c'est principalement dû au guidage par la *vitalité* des propositions de sorties d'objet au sein du filtre, comme l'ont montré les résultats du chapitre 5. La dynamique de la *vitalité* permet à l'utilisateur de choisir une espérance de survie de l'objet à une occultation totale, déterminée par le paramétrage proposé à la section 4.6.3. De plus le choix d'un suivi mené dans l'espace 3D permet une modélisation réaliste de la dynamique des objets, ce qui contraint l'espace de recherche, notamment dans le cas du suivi de véhicules. Par contre, la cohérence des identités est mise en défaut dans deux situations : 1) En cas de forte occultation de deux objets peu contraints dynamiquement (type piétons), et proches l'un de l'autre, rien ne permet de maintenir les bonnes identités. 2) Si un objet cible sort de la scène, puis rentre après une période supérieure à l'espérance de survie de l'objet définie par la dynamique de la *vitalité*, cette dernière n'est d'aucun secours, et on affecte à l'objet cible une nouvelle identité, il n'est donc pas reconnu. La fonction de vraisemblance que nous avons utilisée dans ce manuscrit n'intègre comme descripteur individuel de l'objet, que ses dimensions, ce qui est trop peu discriminant en vidéo-surveillance de véhicules ou de piétons. Telle quelle, la fonction de vraisemblance que nous utilisons ne permet donc pas de surmonter les deux dysfonctionnements décrits ci-dessus. Pour ce faire, une fonction de vraisemblance intégrant un descripteur d'apparences est indispensable, et nous avons mené des expérimentations intégrant un descripteur d'apparence d'objet par histogramme couleur, comme proposé dans (PCJM02). Nous n'avons pas intégré ce descripteur à notre traqueur, ni ces recherches au présent manuscrit, pour les raisons suivantes : 1) Dans les applications de vidéo-surveillance de piétons ou de véhicules, on s'aperçoit que les cibles ont souvent des contenus colorimétriques très proches. Un descripteur couleur est donc insuffisant, et il faudrait le combiner avec d'autres contenus, ce que nous n'avons pas essayé. 2) Du fait des variations de pose et d'illumination des objets, ce descripteur, quel qu'il soit, doit être mis à jour en ligne, ce qui demande l'étude spécifique d'une stratégie de mise à jour. 3) L'intégration de ce descripteur



à la fonction de vraisemblance entraîne des surcoûts de calculs importants. Ce champ de recherche nous semble présenter des potentialités très intéressantes et pourrait faire l'objet d'une étude spécifique.

- ▷ **Fonction d'observation.** Afin de ne pas nous disperser, nous avons choisi de ne pas travailler sur l'observation. Pour cette raison et afin de privilégier la performance temps réel, nous alimentons délibérément le filtre avec une observation pauvre, reposant uniquement sur une segmentation binaire *avant-plan / arrière-plan* basée sur un modèle de l'*arrière-plan* mis à jour en ligne à chaque image et peu gourmande en temps de calcul. Une observation plus riche ne pourrait qu'améliorer les performances de suivi et de classification, en exploitant d'autres attributs tels que les contours, points d'intérêt, gradients, classifieurs...
- ▷ **Fusion multi-caméras.** Dans le cas d'une observation multi-caméra, notre fonction de vraisemblance est vue comme le simple produit des vraisemblances de toutes les observations (section 4.4.4). Ce procédé de fusion des observations doit être revu, car il ne prend pas en compte les disparités d'observabilité d'un même objet  $j$  dans les différentes caméras. Ces disparités peuvent être très élevées, dans le cas où un même objet est occulté dans une caméra et parfaitement observable dans une autre. De plus, cela permettrait d'étendre notre méthode à un système multi-caméra dont les champs de vision ne se recouvrent que partiellement. Une autre extension intéressante est la fusion d'une observation caméra avec une observation télémétrique.
- ▷ **Cartes d'*a priori* d'entrée et de sorties d'objets :** Dans ce travail, nous avons délibérément limité les informations *a priori*, pour concevoir un système de suivi multi-objets générique. L'injection de connaissance *a priori* sur la zone de suivi ne peut qu'améliorer les performances. Des cartes de probabilité d'entrée et de sortie d'objets dans la zone de suivi pourraient par exemple être prises en compte dans les lois *a priori*.
- ▷ **Modélisation des reflets.** Un sol réfléchissant peut pénaliser considérablement la précision du suivi, comme le montrent les résultats de la section 5.4, voire provoquer l'entrée d'objets candidats fantômes. Ce phénomène peut être produit par une chaussée humide, lisse, ou par la présence de surfaces vitrées dans le champ de vision, comme illustré par la figure 1.13 page 18. La modélisation des reflets des objets sur le sol permettrait de minimiser cet effet, à l'image de la modélisation des ombres portées des objets sur la chaussée.
- ▷ **Optimisation des calculs de vraisemblance.** La vraisemblance présentée dans la section 4.4.4, est très coûteuse en calcul car elle nécessite la confrontation de deux images complètes. Ce faisant, nous n'exploitons pas l'opportunité offerte par les propositions de mouvements marginalisées, où un seul objet est bougé ! Nous avons proposé en section 4.4.5 une stratégie de calcul incrémental de cette vraisemblance, que nous n'avons pas encore implémentée, et qui permettra des économies considérables de calculs.
- ▷ **Validité du  $FP\ MCMC^P$ .** Nous avons utilisé le filtre à propositions multiples  $FP\ MCMC^P$  dans le but de respecter les contraintes de temps réel. S'il constitue un moyen d'exploiter les capacités de calculs parallèles des machines actuelles, nous ne pensons pas qu'il réalise une approximation convergeant vers la loi de probabilité *a posteriori* comme le fait le  $FP\ MCMC^1$ . Des investigations complémentaires seraient nécessaires afin de répondre à cette question.
- ▷ **Initialisation de la vitesse des véhicules.** Le modèle dynamique utilisé dans le cas de piétons a été décrit en section 4.2.2, et donne satisfaction. Le modèle dynamique utilisé pour les véhicules a été décrit en section 4.9.2. La dynamique d'un véhicule est fort contrainte par ses capacités d'accélérations longitudinales et transversales. Il est fructueux d'injecter ces contraintes dans la dynamique du véhicule, car cela permet de restreindre la région de l'espace à explorer. Le problème est que cette restriction entre en conflit avec les besoins de la phase d'initialisation des objets candidats, où une forte dynamique est nécessaire car les véhicules cibles entrent dans la zone de suivi avec une vitesse initiale généralement élevée et inconnue. L'application de surveillance d'un carrefour giratoire décrite en section 5.8.3, notamment, doit suivre des véhicules qui peuvent entrer dans la zone de suivi en venant de toutes les directions et à des vitesses très

variables. Un travail reste à mener pour lever ce conflit, par exemple en conditionnant la dynamique à la *vitalité* : forte dynamique en phase d'initialisation d'un nouveau véhicule candidat (lorsque la *vitalité* est faible), puis dynamique réaliste en régime établi (où la *vitalité* est élevée).

**Perspectives : extensions à d'autres applications.** Le système présenté dans ce mémoire reste une méthode générique, laissant ouvertes des potentialités d'extension à d'autres applications :

- ▷ **Extension de la méthode au suivi depuis une caméra embarquée.** Nous avons utilisé une fonction de vraisemblance primaire, uniquement basée sur la segmentation *fond / forme*. La raison principale de ce choix est sa robustesse aux variations de pose et d'éclairement des objets, sa robustesse aux occultations, et la généralité de la méthode, c'est à dire qu'elle peut suivre n'importe quelle catégorie d'objet opaque et (modérément) déformable (type piéton). Cette fonction de vraisemblance ne s'applique évidemment pas en caméra embarquée. Une fonction de vraisemblance basée sur des scores de détection d'objet pourrait prendre la place de l'actuelle fonction de vraisemblance. La contrainte est de trouver un détecteur capable de maintenir un bon score sur un objet partiellement occulté.
- ▷ **Extension de la méthode au suivi d'un corps articulé.** Notre application de suivi multi-objets consiste à suivre des objets aux comportements indépendants lorsqu'ils sont éloignés les uns des autres. Lorsqu'ils évoluent à proximité l'un de l'autre, deux objets adoptent des comportements interdépendants, ce que modélise le potentiel d'interaction répulsif défini en section 4.8. On pourrait donc utiliser la même méthode avec des potentiels attractifs, pour modéliser le comportement d'une structure articulée.



# SUIVI D'OBJETS JOINT CONTRE SUIVI D'OBJETS DISJOINT

Cette annexe propose le suivi expérimental de 10 objets sur images de synthèse, et la quantification de la précision. Ceci permet de comparer les approches de suivi joint et disjoint.

- ▷ Suivi multi-objets disjoint : on instancie un moteur de suivi à chaque nouvel objet. Dans cette approche, chaque moteur opère dans le sous-espace disjoint représentant chaque objet.
- ▷ Suivi multi-objets joint : un unique moteur de suivi est en charge de la totalité des objets. Les objets sont donc représentés dans un espace d'état joint.

## A.1 Evaluation quantitative de la performance du suivi

La précision du suivi est quantifiée par le cumul des distances Euclidiennes entre la position estimée de chaque objet, et sa position vraie. Le suivi d'objets est effectué entre les instants  $t = t_i$  et  $t = t_f$ . Notons  $n = 1$  à  $n = N_t$  les indices des objets à suivre à l'instant  $t$ . Soit  $\mathbf{x}_t^n$  la position de l'objet  $n$  estimée par le système de suivi à l'instant  $t$ , et soit  $\mathbf{c}_t^n$  la position vraie de la cible  $n$  à l'instant  $t$ . L'erreur d'estimation de position s'exprime alors par :

$$\delta_t^n = \sqrt{(\mathbf{c}_t^n - \mathbf{x}_t^n)(\mathbf{c}_t^n - \mathbf{x}_t^n)^T}, \quad (\text{A.1})$$

Les résultats seront présentés sous forme d'histogrammes cumulés des erreurs d'estimation de position. Soit  $\Delta$  un seuil de tolérance d'erreur de position, mesuré en pixels. On définit  $f_{t_i:t_f}^\Delta$ , fréquence mesurée sur la séquence, avec laquelle l'estimation de position de chacun des objets présente une erreur inférieure à  $\Delta$ , selon l'équation [A.2](#)

$$f_{t_i:t_f}^\Delta = \frac{\sum_{t=t_i}^{t=t_f} \sum_{n=1}^{N_t} e_n^t}{\sum_{t=t_i}^{t=t_f} N_t}, \quad (\text{A.2})$$

où  $e_n^t$  est une variable binaire définie par l'équation [A.3](#) :

$$e_n^t = 1 \text{ si } \delta_t^n < \Delta, 0 \text{ sinon} \quad (\text{A.3})$$

## A.2 Résultats

La figure A.1 illustre le suivi de dix cibles rectangulaires en images de synthèse binaire 240 x 240 pixels, objets en blanc, fond en noir. Seuls 60% des pixels de la cible numéro 1 sont classifiés comme *forme*. La segmentation des cibles croît linéairement dans l'ordre des identités croissantes, pour arriver à une segmentation parfaite sur la cible numéro 10 (100% des pixels de la cible 10 classifiés comme *forme*). Les estimations résultant du suivi sont tracées en rouge. Tant qu'il n'y a pas de recouvrement, les deux suivis fonctionnent. A partir de l'image #10009, les cibles #2 et #7 se recouvrent. Le suivi disjoint (ligne du haut), défaille car le filtre chargé du suivi de la cible #2 est attiré vers la cible #7, mieux segmentée. Dès lors, la cible #2 n'est plus suivie ! Le phénomène se reproduit à l'image #10060, où le filtre chargé du suivi de la cible #5 la lâche car il est attiré vers la cible #10, mieux segmentée. La suite présentée sur la figure A.2, montre les pertes successives des cibles #8 (image #10079), #1 (image #10119) et #3 (image #10168), toujours au profit d'une cible mieux segmentée. La courbe de  $f^{\Delta}$  pour le suivi disjoint (vert) se dégradent à mesure de ces pertes de cibles. Le suivi joint, pendant ce temps, garde un excellente performance, comme attesté par sa courbe de  $f^{\Delta}$  (rouge), montrant que 95% des estimations présentent une erreur de position inférieure à 2 pixels !

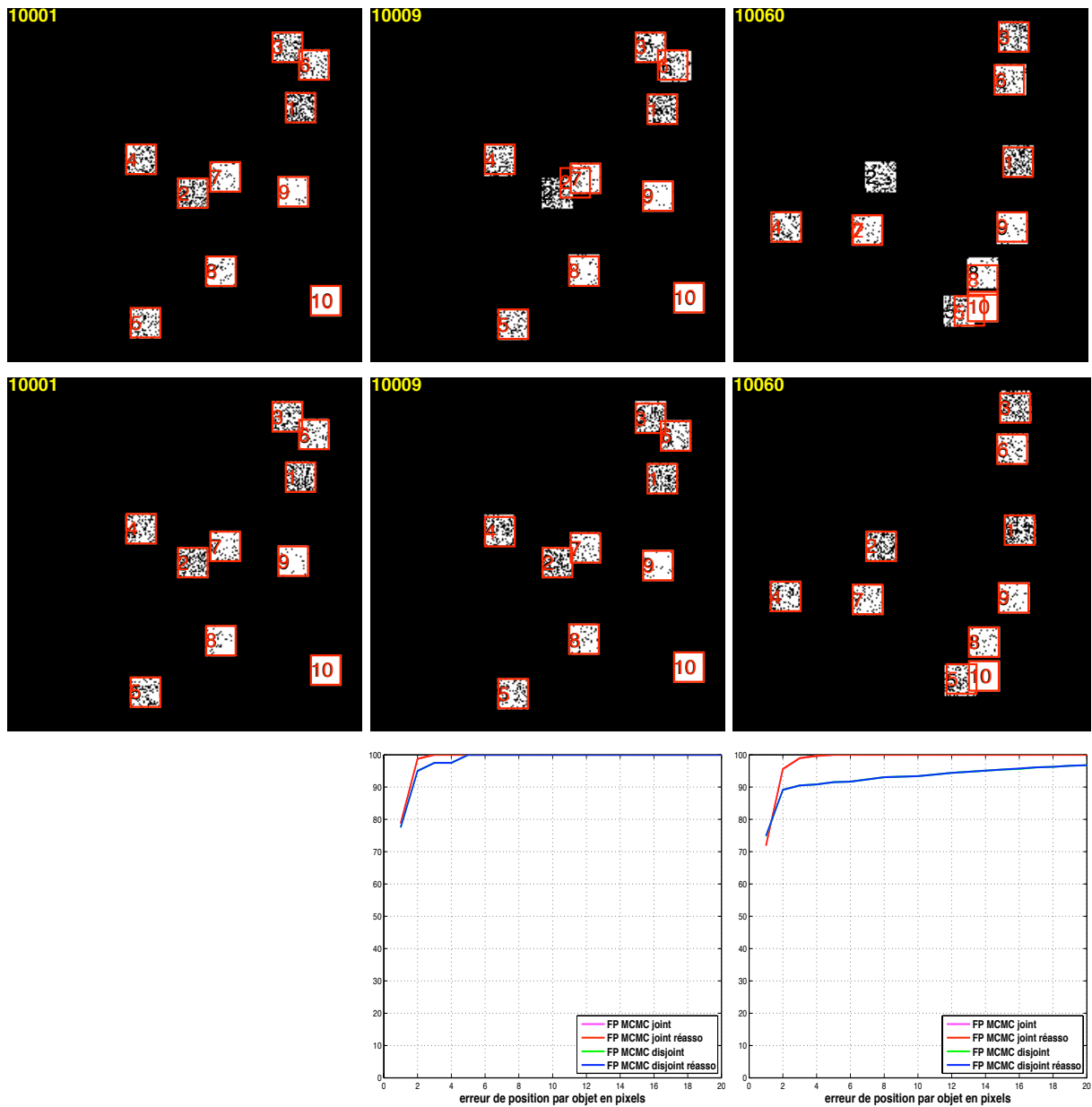


FIGURE A.1 – Suivi de 10 cibles rectangulaires de synthèse sur image binaire  $240 \times 240$  pixels. La segmentation des cibles est linéairement croissante de 60% (cible #1) à 100% (cible #10). Estimations en rouge. Haut : suivi disjoint, un filtre instancié pour chaque cible, 20 particules par filtre. Milieu : suivi joint de toutes les cibles par un seul filtre, 200 particules. Bas : histogrammes cumulés :  $f^\Delta$  en fonction de  $\Delta$  (en abscisses, variant de 1 à 20 pixels). En rouge : suivi par filtre joint. En vert : suivi par filtre disjoint. En bleu : suivi par filtre disjoint, suivi d'une réassociation à la cible la plus proche.

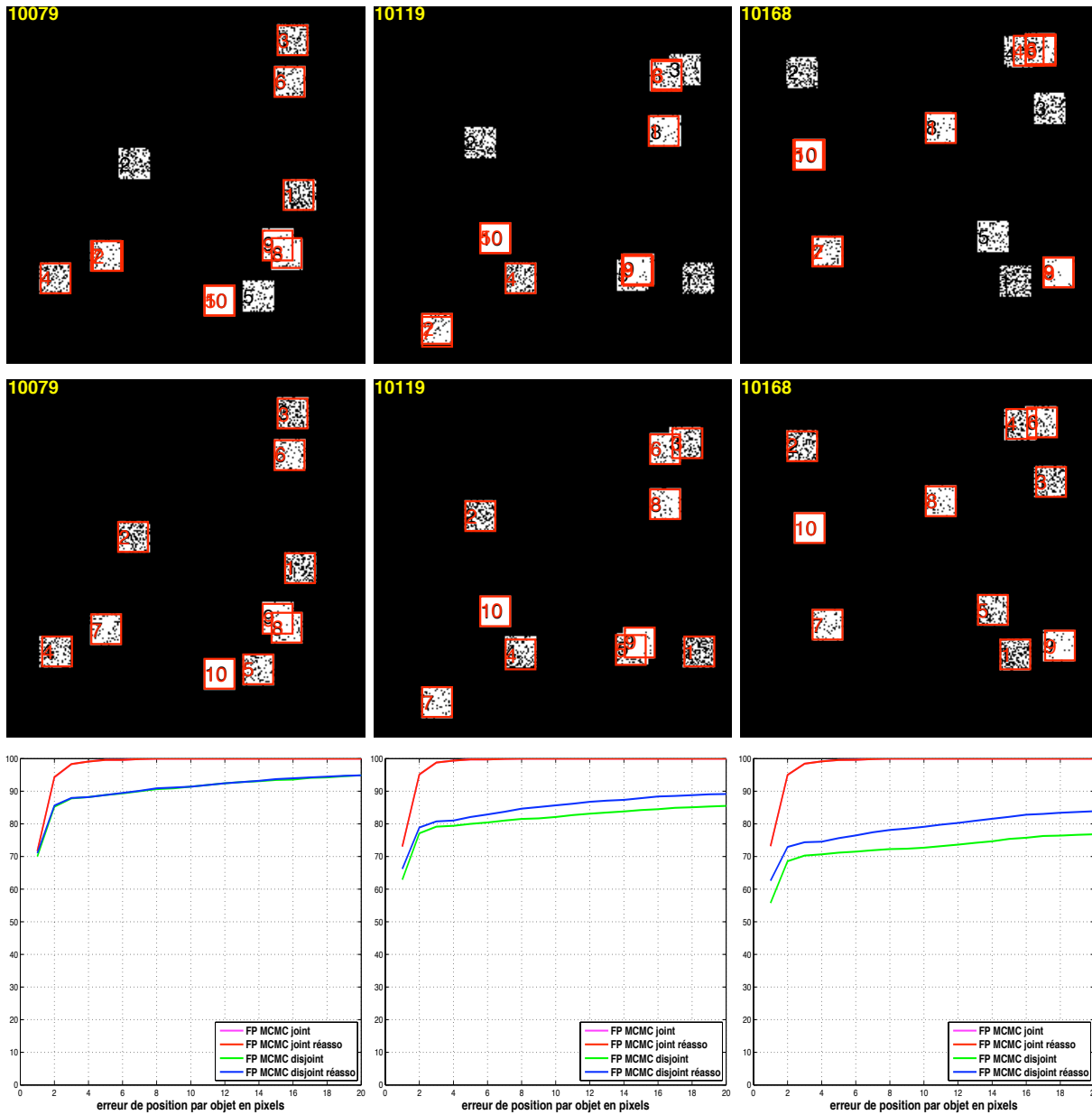


FIGURE A.2 – Suite du suivi de 10 cibles rectangulaires de synthèse sur image binaire  $240 \times 240$  pixels. La segmentation des cibles est linéairement croissante de 60% (cible #1) à 100% (cible #10). Estimations en rouge. Haut : suivi disjoint, un filtre instancié pour chaque cible, 20 particules par filtre. Milieu : suivi joint de toutes les cibles par un seul filtre, 200 particules. Bas : histogrammes cumulés :  $f^\Delta$  en fonction de  $\Delta$  (en abscisses, variant de 1 à 20 pixels). En rouge : suivi par filtre joint. En vert : suivi par filtre disjoint. En bleu : suivi par filtre disjoint, suivi d'une réassociation à la cible la plus proche.

# ÉCHANTILLONNAGE PRÉFÉRENTIEL

## B.1 Échantillonnage Préférentiel (ou *Importance Sampling, IS*)

Soit  $\mathbf{X}$  une variable aléatoire, et  $p(\mathbf{X})$  sa loi de probabilité. Un échantillonneur a pour fonction de simuler des tirages aléatoires selon cette loi, de façon à l'approximer par un ensemble d'échantillons discrets. On a vu dans la section 2.2, que cette loi peut être approximée par une somme de  $N$  échantillons pondérés :

$$p(\mathbf{X}) \approx \sum_{n=1}^N \pi^n \delta(\mathbf{X} - \mathbf{X}^n), \text{ que l'on note également : } p(\mathbf{X}) \approx \{\mathbf{X}^n, \pi^n\}_{n=1}^N \quad (\text{B.1})$$

Cependant, lorsqu'on n'a pas de modèle simple de  $p(\mathbf{X})$ , le tirage est difficile. Une solution très employée est le recours à l'*Échantillonnage Préférentiel* (ou *Importance Sampling, IS*). Cette méthode consiste à confier le tirage des échantillons à une loi plus simple  $q(\mathbf{X})$  dite *loi de proposition* ou *loi d'importance*. Pour tenir compte du fait que les échantillons ne sont pas tirés selon  $p(\mathbf{X})$  mais selon  $q(\mathbf{X})$ , chaque échantillon est pondéré par un *poids d'importance*, calculé selon :

$$\pi^n = \frac{p(\mathbf{X}^n)}{q(\mathbf{X}^n)}, \quad (\text{B.2})$$

où  $p(\mathbf{X}^n)$  et  $q(\mathbf{X}^n)$  sont les évaluations respectives des lois  $p(\mathbf{X})$  et  $q(\mathbf{X})$  sur l'échantillon  $\mathbf{X}^n$ . L'Échantillonnage préférentiel d'une loi quelconque en variable monodimensionnelle, est illustré par la figure B.1. Cette figure montre que le choix de la loi de proposition est déterminant pour obtenir une approximation correcte. Le graphe du haut montre en effet que si la loi d'importance  $q(\mathbf{X})$  est mal adaptée à la loi cible  $p(\mathbf{X})$ , *i.e.* présente une faible densité dans des régions où la densité de la loi cible  $p(\mathbf{X})$  est élevée, peu d'échantillons de fort poids représenteront cette région, l'échantillonnage est donc mauvais.

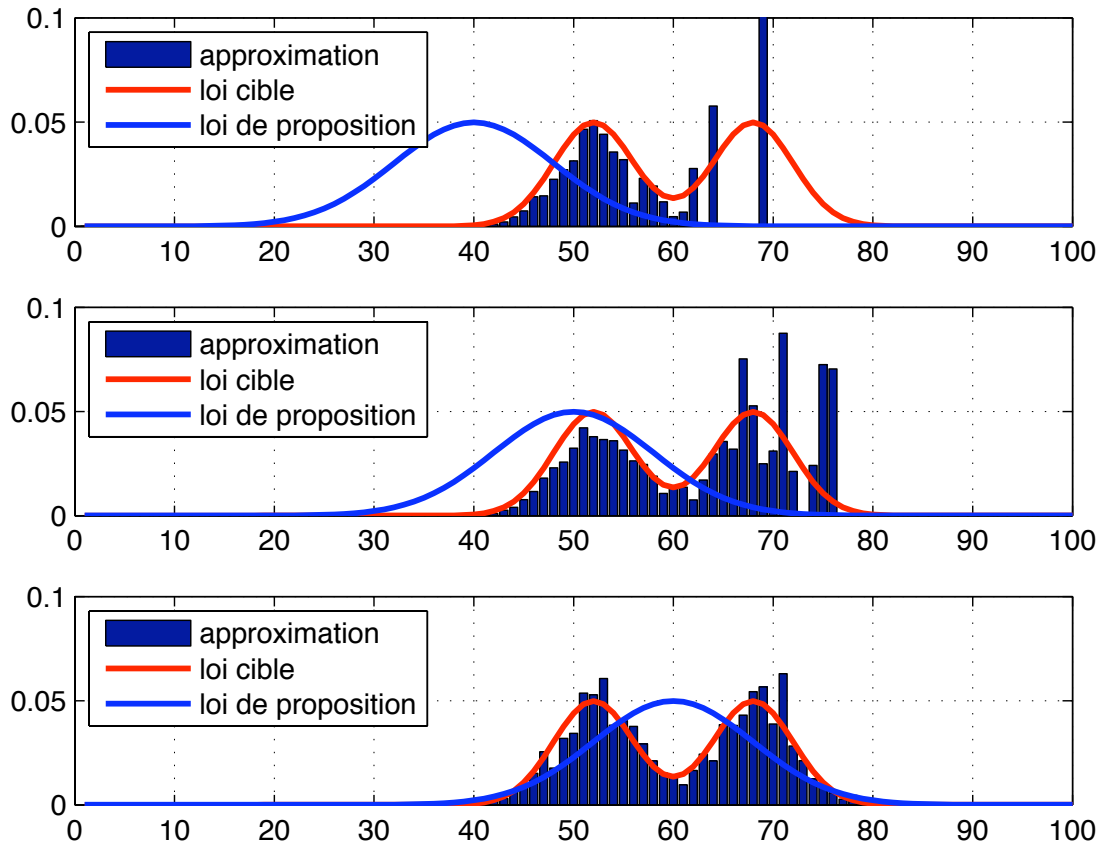


FIGURE B.1 – Approximation de la loi de probabilité  $p(\mathbf{X}) = 0.5\mathcal{N}(52, 8^2) + 0.5\mathcal{N}(68, 8^2)$  (courbe continue rouge) par échantillonnage préférentiel (histogramme) avec  $N = 1000$  échantillons. La loi d'importance est représentée par la courbe continue bleue. En haut : loi d'importance  $q(\mathbf{X}) = \mathcal{N}(40, 8^2)$ . Milieu :  $q(\mathbf{X}) = \mathcal{N}(50, 8^2)$ . Bas :  $q(\mathbf{X}) = \mathcal{N}(60, 8^2)$



## B.2 Échantillonnage Préférentiel Séquentiel (ou *Sequential Importance Sampling, SIS*)

Destiné au suivi de phénomènes dynamiques, cet algorithme appartient au domaine des Filtres Particulaires. Il consiste à propager séquentiellement un nuage de particules approximant à chaque instant la loi de probabilité *a posteriori* de l'état du système connaissant les mesures. Soit  $\mathbf{X}_t$  l'état de la variable aléatoire  $\mathbf{X}$  à l'instant  $t$ , et  $p(\mathbf{X}_t)$  sa loi de probabilité. Notons  $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$  la loi de probabilité *a posteriori* de l'état à l'instant  $t$ , connaissant l'historique des mesures  $\mathbf{Z}_{1:t}$ . Supposons que l'on dispose d'une approximation de  $p(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1})$  par  $N$  échantillons discrets pondérés  $\{\mathbf{X}_{t-1}^n, \pi_{t-1}^n\}_{n=1}^N$  :

$$p(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1}) \approx \sum_{n=1}^N \pi_{t-1}^n \delta(\mathbf{X}_{t-1} - \mathbf{X}_{t-1}^n), \quad (\text{B.3})$$

où  $\delta$  est l'impulsion de Dirac, et  $\pi_{t-1}^n$  est le poids du  $n^{\text{ième}}$  échantillon,  $n \in 1 \dots N$ , tel que  $\sum_{n=1}^N \pi_{t-1}^n = 1$ . L'algorithme *SIS* permet d'en déduire récursivement une approximation de  $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$ , loi de probabilité *a posteriori* de  $\mathbf{X}_t$  connaissant l'historique des mesures  $\mathbf{Z}_{1:t}$ , par  $N$  nouveaux échantillons discrets pondérés  $\{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N$  :

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t}) \approx \sum_{n=1}^N \pi_t^n \delta(\mathbf{X}_t - \mathbf{X}_t^n), \quad (\text{B.4})$$

où les poids sont remis à jour selon l'équation :

$$\pi_t^n = \pi_{t-1}^n \frac{p(\mathbf{Z}_t|\mathbf{X}_t^n)p(\mathbf{X}_t^n|\mathbf{X}_{t-1}^n)}{q(\mathbf{X}_t^n|\mathbf{X}_{t-1}^n, \mathbf{Z}_t)}, \quad (\text{B.5})$$

où  $p(\mathbf{Z}_t|\mathbf{X}_t^n)$  est la vraisemblance de l'observation  $\mathbf{Z}_t$  connaissant l'état candidat  $\mathbf{X}_t^n$ , tandis que  $p(\mathbf{X}_t^n|\mathbf{X}_{t-1}^n)$  est la loi dynamique de l'état, enfin  $q(\mathbf{X}_t^n|\mathbf{X}_{t-1}^n, \mathbf{Z}_t)$  est la loi de proposition d'un état  $\mathbf{X}_t^n$  connaissant l'état précédent  $\mathbf{X}_{t-1}^n$  et l'observation  $\mathbf{Z}_t$ . Si on se contente d'une loi de proposition « aveugle », *i.e.* telle que  $q(\mathbf{X}_t^n|\mathbf{X}_{t-1}^n, \mathbf{Z}_t) = q(\mathbf{X}_t^n|\mathbf{X}_{t-1}^n)$ , il est raisonnable de choisir la loi dynamique comme loi de proposition. Dans ce cas, l'équation B.5 se simplifie selon :

$$\pi_t^n = \pi_{t-1}^n p(\mathbf{Z}_t|\mathbf{X}_t^n) \quad (\text{B.6})$$

L'efficacité de cet algorithme est compromise par le phénomène de dégénérescence des particules : après quelques itérations, presque toutes les particules portent des poids négligeables. Il a en effet été montré que la variance des poids ne peut qu'augmenter avec le temps. L'algorithme tend à approximer la loi de probabilité par une seule particule. Ce phénomène de dégénérescence des particules est illustré sur la colonne de gauche de la figure , où dès l'instant  $t = 20$ , deux particules de poids forts représentent quasiment à elles seules la loi de probabilité. Une façon d'éviter cette dégénérescence est de rééchantillonner les particules, dès que la variance de leurs poids devient excessive.

**Rééchantillonnage des particules.** On dispose initialement d'un jeu de particules pondérées  $\{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N$  pour approximer la loi de probabilité *a posteriori*  $p(\mathbf{X}_t|\mathbf{Z}_t)$  :  $p(\mathbf{X}_t|\mathbf{Z}_t) \approx \{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N$ . Les poids sont normalisés tels que  $\sum_{n=1}^N \pi_t^n = 1$ . Le rééchantillonnage des particules consiste à générer un nouveau jeu de particules non pondérées  $\{\mathbf{X}_t^{n*}, 1/N\}_{n=1}^N$ , approximant lui aussi la loi de probabilité *a posteriori*  $p(\mathbf{X}_t|\mathbf{Z}_t)$  :  $p(\mathbf{X}_t|\mathbf{Z}_t) \approx \{\mathbf{X}_t^{n*}, 1/N\}_{n=1}^N$ . Chaque nouvelle particule est générée par tirage avec remise dans le jeu de particule initial, avec pour probabilité le poids correspondant :  $P(\mathbf{X}_t^{n*} = \mathbf{X}_t^m) = \pi_t^m$ .

## B.3 Rééchantillonnage Préférentiel Séquentiel (ou *Sequential Importance Resampling, SIR*)

L'algorithme *SIR* est une variante de l'algorithme *SIS*, dans laquelle le rééchantillonnage des particules est conduit à chaque pas temporel. Le résultat est illustré sur la colonne du milieu de la figure , montrant que cette

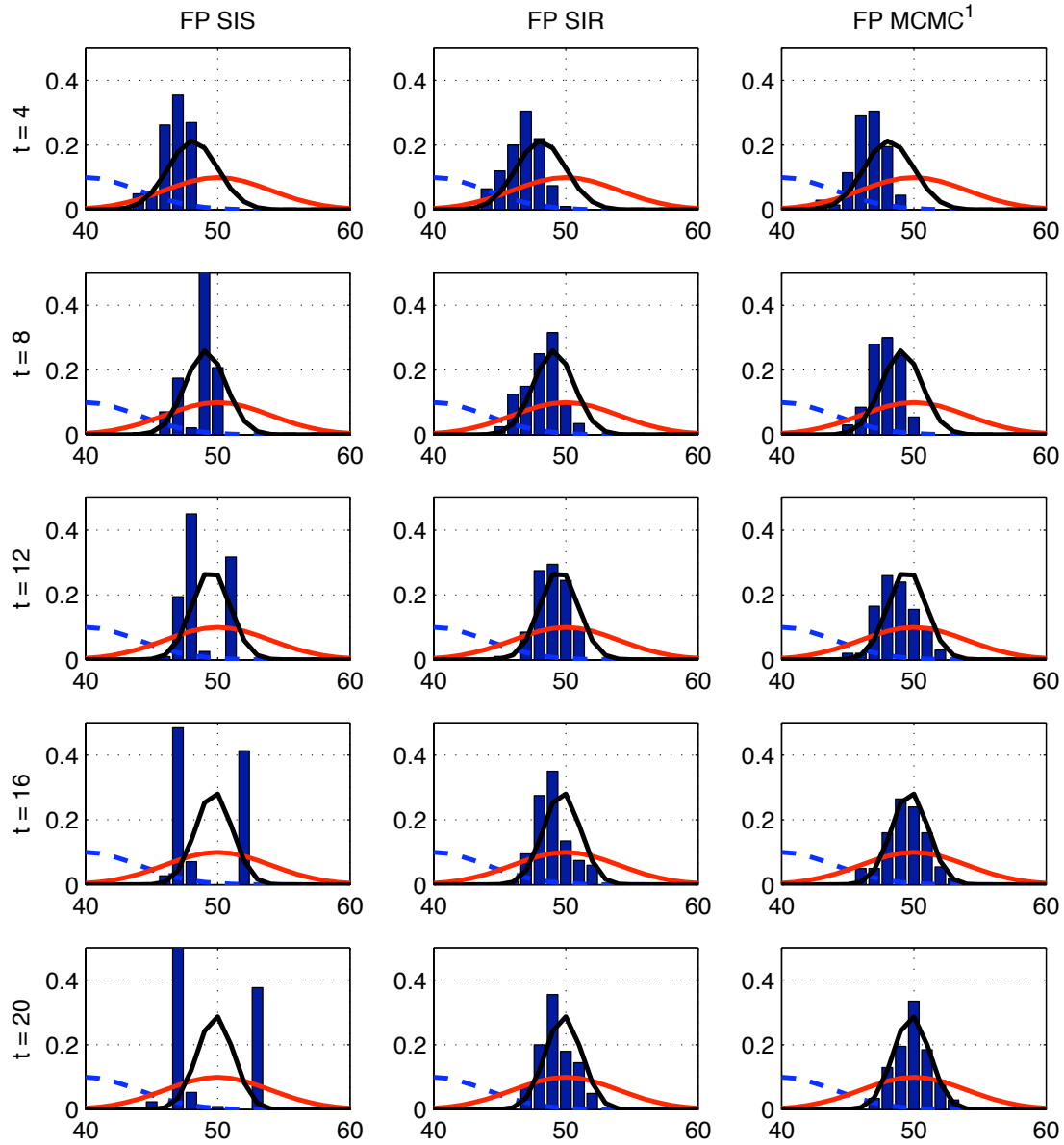


FIGURE B.2 – Filtrage d’une densité par *SIS* (à gauche), *SIR* (colonne du milieu) et *FP MCMC<sup>1</sup>* (à droite). En pointillés bleus,  $p(\mathbf{X}_0) = \mathcal{N}(40, 4^2)$ , loi de probabilité *a posteriori* initiale à  $t=0$ . En trait continu rouge, la fonction de vraisemblance stationnaire de l’instant  $t = 1$  jusqu’à  $t = 20$  :  $p(\mathbf{Z}_t|\mathbf{X}_t) = \mathcal{N}(50, 4^2)$ . En trait continu noir, la loi de probabilité *a posteriori* délivrée par un filtre Kalman à chaque instant. L’approximation échantillonnée de la loi de probabilité *a posteriori* à chaque instant est tracée sous forme d’histogramme. Pour les trois filtres particuliers, on utilise  $N = 200$  particules. Le modèle dynamique est identique pour tous les filtres :  $p(\mathbf{X}_t|\mathbf{X}_{t-1}) = \mathcal{N}(0, 0.5^2)$ .

stratégie permet d'éviter le phénomène de dégénérescence des particules rencontré avec le *SIS*.



## PERFORMANCES DES FILTRES PARTICULAIRES : COMPLÉMENTS

Au chapitre 3, nous avons présenté les performances des filtres selon la divergence de Kullback-Leibler. Cette annexe complète ces résultats en présentant les performances des filtres selon les deux autres critères : 1) La déviation moyenne par dimension entre  $\mu_{FP}$ , moyenne du nuage de particules rééchantillonnées par le Filtre Particulaire, et  $\mu_{FK}$ , moyenne estimée par Filtre de Kalman. 2) Le taux d'unicité des l'échantillons  $\nu_u$ , défini par l'équation (3.5).

Sur les graphes de résultats, ces quatre filtres sont désignés comme suit :

- ▷ *FP SIR* pour le Filtre Particulaire SIR (Sample Importance Resampling)
- ▷ *FP partition* pour le Filtre Particulaire partitionné
- ▷ *FP MCMC<sub>d</sub><sup>1</sup>* pour le Filtre Particulaire MCMC, l'indice  $d$  indiquant la dimension du sous-espace marginalisant les propositions, c'est à dire le nombre de composantes simultanément modifiées à chaque itération. Cet indice peut prendre toute valeur entière comprise entre 1 (on ne bouge alors qu'une composantes à la fois) et  $D$  (on bouge alors toutes les composantes simultanément). L'exposant 1 désigne le Filtre Particulaire MCMC classique, où une seule proposition est générée et évaluée à chaque itération.
- ▷ *FP MCMC<sub>d</sub><sup>p</sup>* pour le Filtre Particulaire MCMC, l'indice  $d$  indiquant le nombre de dimensions simultanément modifiées à chaque itération. L'exposant  $p$  désigne l'extension parallélisée du Filtre Particulaire MCMC, où  $p$  propositions sont simultanément générées et évaluées par  $p$  processeurs à chaque itération.

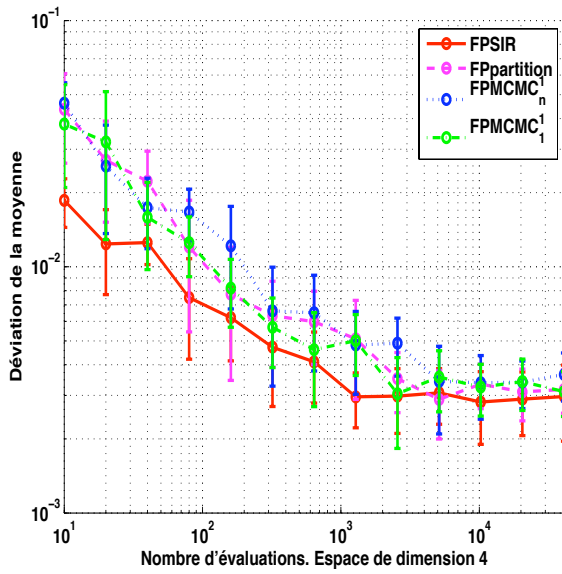


FIGURE C.1 – Déviation moyenne par dimension, de la moyenne estimée par les Filtres Particulaires à la moyenne estimée par Filtre de Kalman, en dimension 4. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

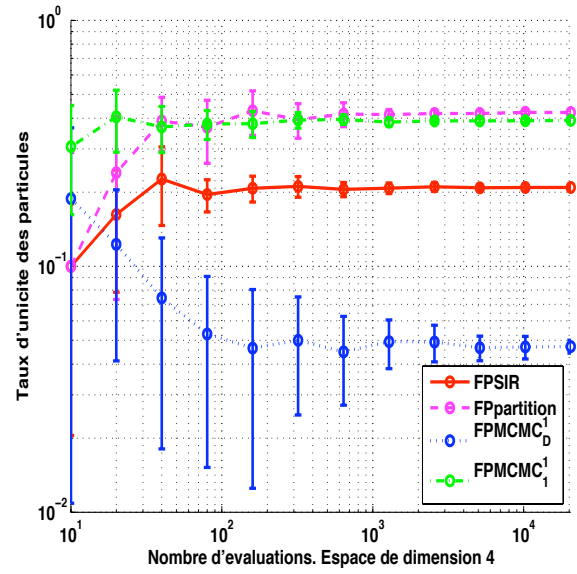


FIGURE C.2 – Taux d'unicité des particules en dimension 4. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

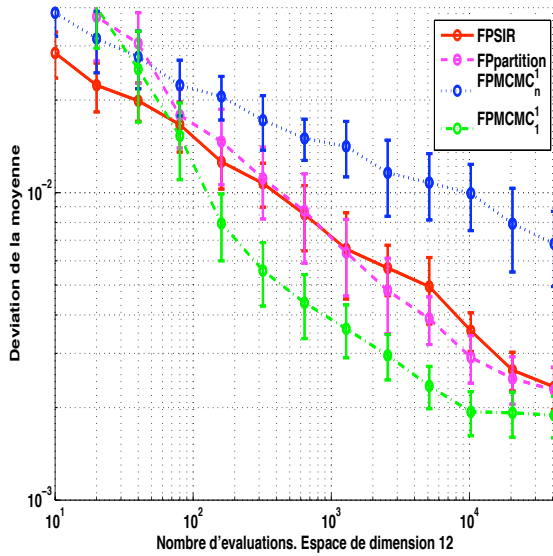


FIGURE C.3 – Déviation moyenne par dimension, de la moyenne estimée par les Filtres Particulaires à la moyenne estimée par Filtre de Kalman, en dimension 12. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

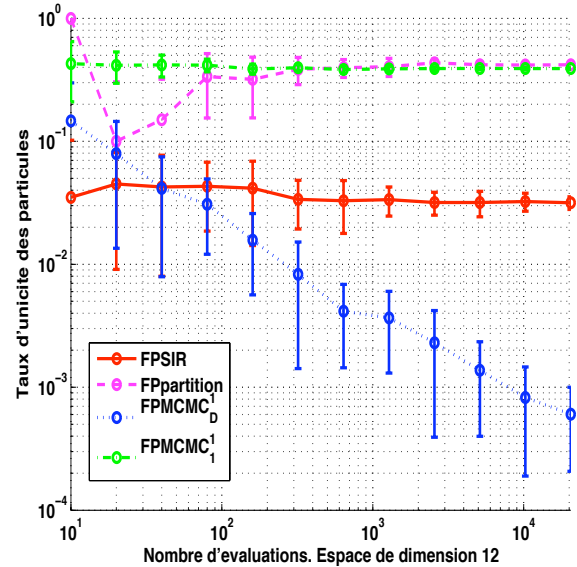


FIGURE C.4 – Taux d'unicité des particules en dimension 12. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.



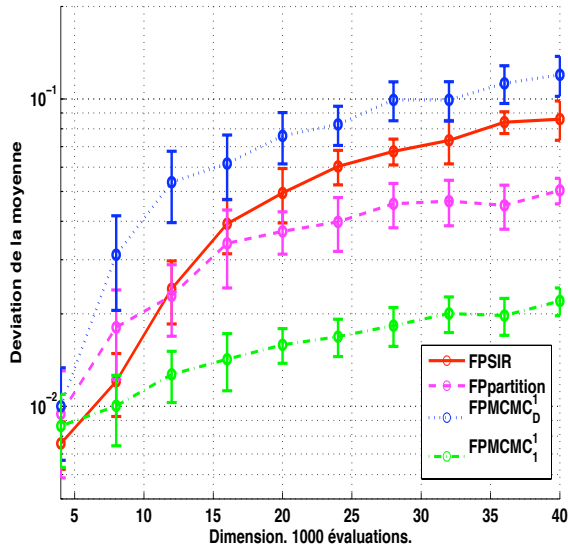


FIGURE C.5 – Déviation moyenne par dimension, de la moyenne estimée par les Filtrés Particulaires à la moyenne estimée par Filtre de Kalman, 1000 évaluations. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

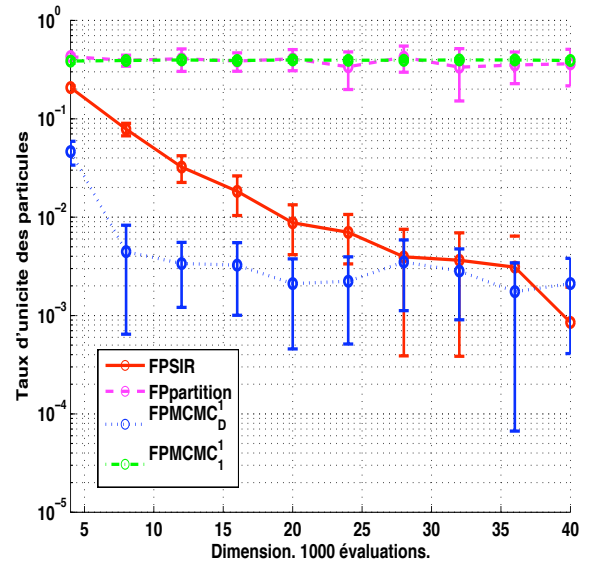


FIGURE C.6 – Taux d'unicité des particules, 1000 évaluations.

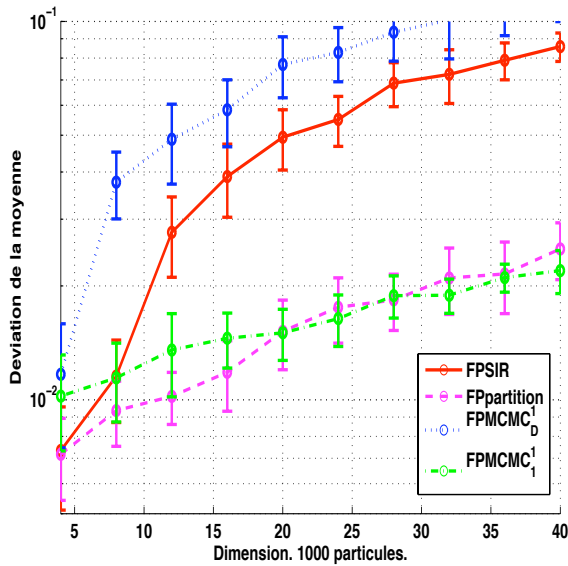


FIGURE C.7 – Déviation moyenne par dimension, de la moyenne estimée par les Filtrés Particulaires à la moyenne estimée par Filtre de Kalman, 1000 particules. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

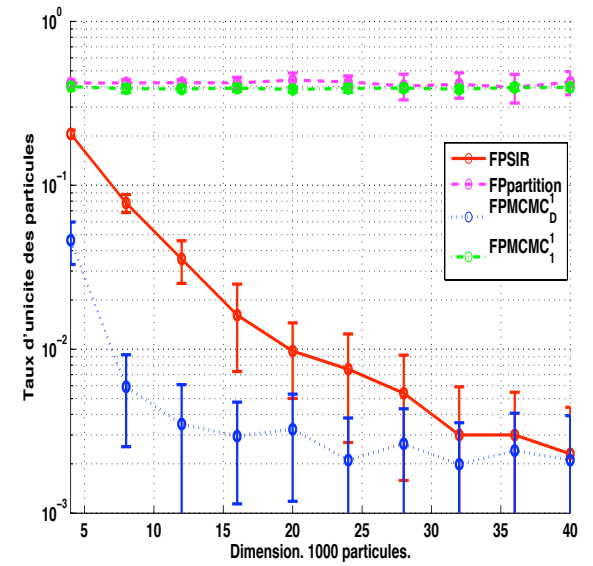


FIGURE C.8 – taux d'unicité des 1000 particules

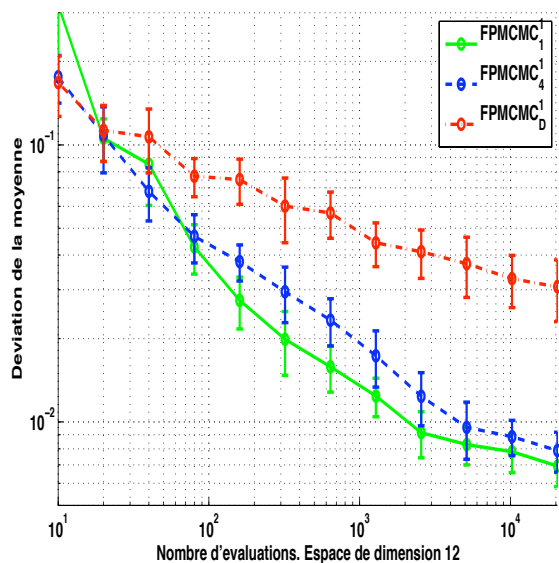


FIGURE C.9 – Déviation moyenne par dimension, de la moyenne estimée par les *FP MCMC* à la moyenne estimée par Filtre de Kalman, 1000 évaluations.

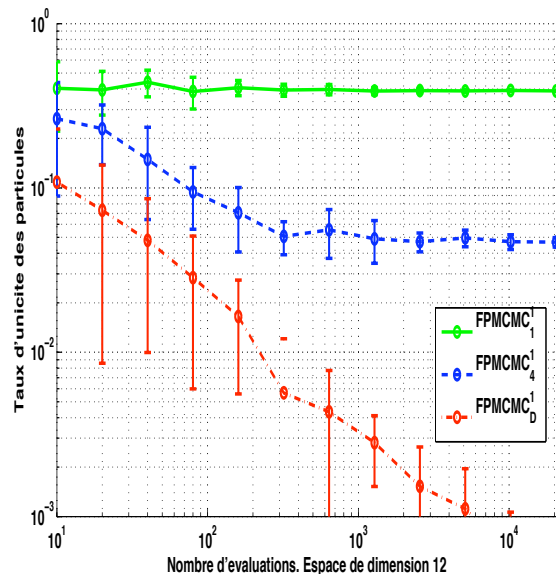


FIGURE C.10 – Taux d'unicité des particules des *FP MCMC*, 1000 évaluations.

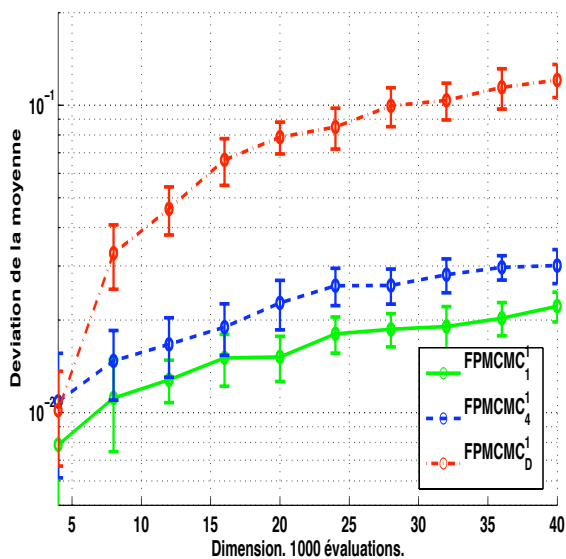


FIGURE C.11 – Déviation moyenne par dimension, de la moyenne estimée par les *FP MCMC* à la moyenne estimée par Filtre de Kalman, 1000 évaluations.

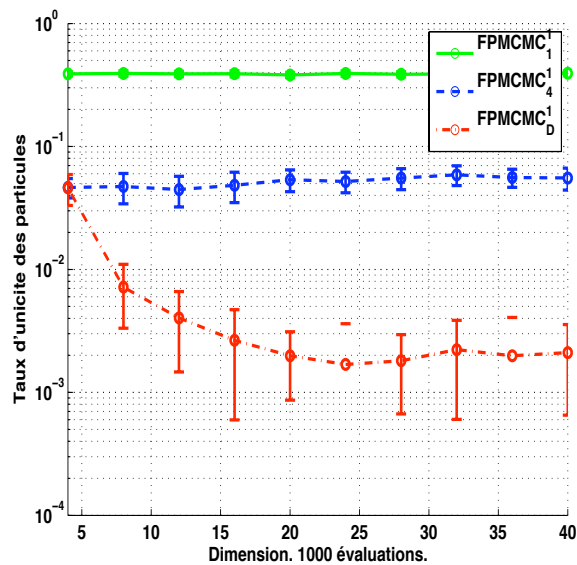


FIGURE C.12 – Taux d'unicité des particules des *FP MCMC*, 1000 évaluations.

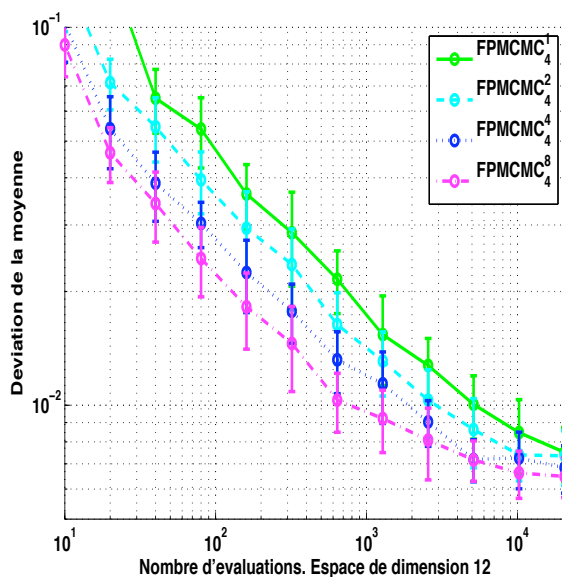


FIGURE C.13 – Déviation moyenne par dimension de la moyenne estimé par les *FP MCMC*, par rapport à celle estimée par Filtre de Kalman, 1000 évaluations. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

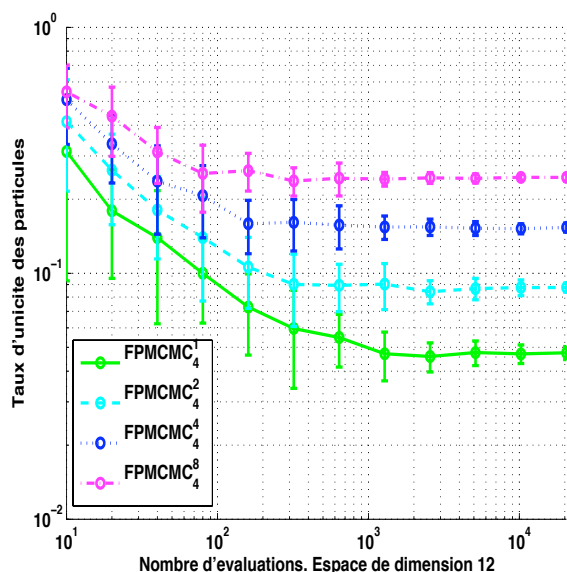


FIGURE C.14 – Unicité des particules.

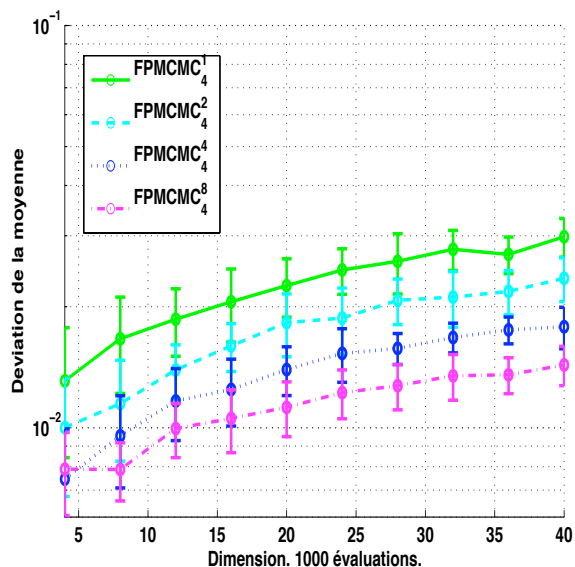


FIGURE C.15 – Déviation moyenne par dimension de la moyenne estimé par les *FP MCMC*, par rapport à celle estimée par Filtre de Kalman, avec 1000 évaluations. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.

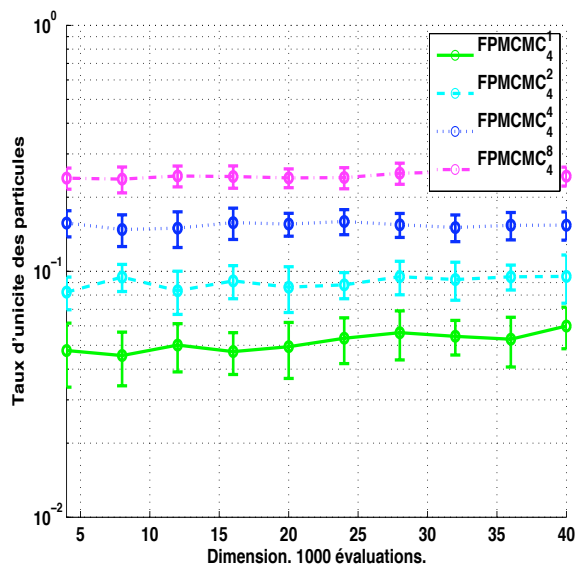


FIGURE C.16 – Unicité des particules, 1000 évaluations. Les barres d'erreurs ont pour demi-hauteur l'écart-type de cette distance sur 20 répétitions.



---

## NOTATIONS ET CONVENTIONS

Tout au long de ce manuscrit, les vecteurs sont représentés par des lettres en gras alors que les variables monodimensionnelles sont en fonte normale. Les acronymes suivants sont précédés de leur équivalent français lorsqu'ils sont issus de la langue anglaise.

## D.1 Acronymes

<i>IV</i> .....	Véhicules Intelligents, Intelligent Vehicles.
<i>ITS</i> .....	Systèmes de Transports Intelligents, Intelligent Transportation Systems.
<i>ADAS</i> .....	Systèmes d'Assistance Avancés au Conducteur, Advanced Driver Assistance Systems.
<i>LIDAR</i> .....	Télémètre par LASER, Light Detection and Ranging.
<i>FP</i> .....	Filtre Particulaire, Particle Filter.
<i>SIS</i> .....	Échantillonnage pondéré séquentiel, Sequential Importance Sampling.
<i>SIR</i> .....	Échantillonnage pondéré séquentiel avec rééchantillonnage, Sequential Importance Resampling.
<i>FP SIR</i> .....	Filtre Particulaire basé sur un Échantillonnage <i>SIR</i> , Sequential Importance Sampling Particle Filter.
<i>FP Partition</i> .....	Filtre Particulaire Partitionné, Partitionned Particle Filter.
<i>MCMC</i> .....	Méthode de Monte-Carlo par Chaîne de Markov, Markov Chain Monte Carlo.
<i>FP MCMC</i> .....	Filtre Particulaire échantillonné par Méthode de Monte-Carlo par Chaîne de Markov, Markov Chain Monte Carlo Particle Filter.
<i>FP MCMC<sub>1</sub></i> .....	Filtre Particulaire <i>MCMC</i> , dans lequel chaque particule $n$ de la chaîne est proposée en ne perturbant qu'une des composantes de la particule $n - 1$ .
<i>FP MCMC<sub>D</sub></i> .....	Filtre Particulaire <i>MCMC</i> , dans lequel chaque particule $n$ de la chaîne est proposée en perturbant simultanément toutes les $D$ composantes de la particule $n - 1$ .
<i>FP MCMC<sub>d</sub></i> .....	Filtre Particulaire <i>MCMC</i> , dans lequel chaque particule $n$ de la chaîne est proposée en perturbant simultanément $d \leq D$ composantes de la particule $n - 1$ ( $D$ est la dimension de l'espace d'état).
<i>FP MCMC<sup>P</sup></i> .....	Filtre Particulaire <i>MCMC</i> , dans lequel chaque particule de la chaîne est tirée parmi $P$ propositions générées en parallèle.
<i>FP MCMC<sub>d</sub><sup>P</sup></i> .....	Filtre Particulaire <i>MCMC</i> , à $P$ propositions parallèles, chacune perturbant $d$ composantes du vecteur d'état.
<i>FP RJ-MCMC</i> ..	Filtre Particulaire échantillonné par Méthode de Monte-Carlo par Chaîne de Markov à Mouvements Reversibles, Reversible Jump Markov Chain Monte Carlo Particle Filter.
<i>MAP</i> .....	Maximum A Posteriori.
<i>KDE</i> .....	Estimation de densité par la méthode du noyau, Kernel Density Estimation.
<i>MOT</i> .....	Suivi Multi-Objets, Multiple Object Tracking.
<i>MOTS</i> .....	Suivi Multi-Objets avec modélisation des Ombres,
<i>MOTC<sup>k</sup></i> .....	Suivi et Classification Multi-Objets en $k$ classes,
<i>MOTC<sup>k</sup>S</i> .....	Suivi et Classification Multi-Objets en $k$ classes, modélisation des Ombres.



## D.2 Notations

$p$ .....	Loi de probabilité d'une variable continue.
$P$ .....	Probabilité d'une variable discrète.
$t$ .....	Lettre utilisée en indice pour désigner le temps discret.
$j$ .....	Lettre utilisée en exposant pour désigner un des $J$ objets.
$n$ .....	Lettre utilisée en exposant pour désigner une des $N$ particules.
$\mathbf{X}$ .....	Vecteur d'état Markovien à estimer.
$\mathbf{Z}$ .....	Vecteur de mesures.
$\mathbf{X}_t$ .....	État joint de la configuration multi-objets à l'instant $t$ .
$\mathbf{X}_t^n$ .....	Particule $n$ décrivant l'état joint multi-objets à l'instant $t$ .
$\mathbf{x}_t^j$ .....	État de l'objet $j$ à l'instant $t$ .
$\mathbf{x}_t^{j,n}$ .....	État de l'objet $j$ décrit par la particule $n$ à l'instant $t$ .
$\mathcal{N}(\mu, \Sigma)$ .....	Distribution Gaussienne de moyenne $\mu$ et de covariance $\Sigma$ .
$\mathcal{U}(a, b)$ .....	Distribution uniforme sur l'intervalle $[a, b]$ .
$\delta$ .....	Impulsion de Dirac.
$\mathbf{A}^T$ .....	Transposée de la matrice $\mathbf{A}$
$\mathbf{A}^{-1}$ .....	Inverse de la matrice $\mathbf{A}$
$p(\mathbf{Z} \mathbf{X})$ .....	Loi de vraisemblance de l'observation $\mathbf{Z}$ étant donné l'état $\mathbf{X}$ .
$p(\mathbf{X}_t \mathbf{Z}_{1:t-1})$ ....	Loi de probabilité <i>a priori</i> de l'état $\mathbf{X}_t$ étant données les observations passées $\mathbf{Z}_{1:t-1}$ .
$p(\mathbf{X}_t \mathbf{Z}_{1:t})$ .....	Loi de probabilité <i>a posteriori</i> de l'état $\mathbf{X}_t$ étant données les observations passées et courante $\mathbf{Z}_{1:t}$ .
$p(\mathbf{X}_t \mathbf{X}_{t-1})$ ....	Modèle d'évolution dynamique.
$D_{\text{KL}}(p  q)$ .....	divergence de Kullback-Leibler de la loi de probabilité $q$ par rapport à la loi de probabilité $p$ .
$\mathcal{A} \cap \mathcal{B}$ .....	Intersection de 2 ensembles $\mathcal{A}$ et $\mathcal{B}$ (ensemble des éléments qui sont dans $\mathcal{A}$ et dans $\mathcal{B}$ ).
$\mathcal{A} \cup \mathcal{B}$ .....	Réunion de 2 ensembles $\mathcal{A}$ et $\mathcal{B}$ (ensemble des éléments qui sont dans $\mathcal{A}$ ou dans $\mathcal{B}$ ).
$\mathcal{A} \subset \mathcal{B}$ .....	Inclusion de l'ensemble $\mathcal{A}$ dans l'ensemble $\mathcal{B}$ .
$\bar{\mathcal{A}}$ .....	Complément de $\mathcal{A}$ (ensemble des éléments qui ne sont pas dans $\mathcal{A}$ ).
$\mathcal{A} \setminus a$ .....	Ensemble $\mathcal{A}$ privé de l'élément $a$ .



# Bibliographie

- [AMGC02] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2) :174–188, February 2002. 26
- [ANN01] A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte-Carlo Methods in practice*. Springer, 2001. 57
- [BC08a] François Bardet and Thierry Chateau. MCMC particle filter for real-time visual tracking of vehicles. In *International IEEE Conference on Intelligent Transportation Systems*, pages 539 – 544, 2008. 17, 69, 81
- [BC08b] François Bardet and Thierry Chateau. Performances comparées de rééchantillonnage pour filtres de monte-carlo. In *Reconnaissance des Formes et Intelligence Artificielle*, 2008. 52
- [BCR09a] François Bardet, Thierry Chateau, and Datta Ramadasan. Illumination aware MCMC particle filter for long-term outdoor multi-object simultaneous tracking and classification. In *International Conference on Computer Vision*, volume 1, 2009. 69
- [BCR09b] François Bardet, Thierry Chateau, and Datta Ramadasan. Real-time multi-object tracking with few particles : A parallel extension of MCMC algorithm. In *International Conference on Computer Vision Theory and Applications*, volume 1, pages 456–463, 2009. 69
- [BCR09c] François Bardet, Thierry Chateau, and Datta Ramadasan. Unifying real-time multi-vehicle tracking and categorization. In *Intelligent Vehicle Symposium*, volume 1, 2009. 69, 104
- [BEBV08] T. Bouwmans, F. El Baf, and B. Vachon. Background modeling using mixture of gaussians for foreground detection - a survey. *Recent Patents on Computer Science*, 1(3) :219–237, 2008. 15
- [BRB<sup>+</sup>08] T. Bragatto, G. Ruas, V. Benso, M. Lamar, D. V.Aldigueri, G.L. Teixeira, and Y. Yamashita. A new approach to multiple vehicle tracking in intersections using harris corners and adaptive background subtraction. In *Int. IEEE Intelligent Vehicle Symposium*, pages 548–553, 2008. 3
- [BvHH09] Michael Beetz and Nicolai v. Hoyningen-Huene. Rao-blackwellized resampling particle filter for real-time player tracking in sports. In *International Conference on Computer Vision Theory and Applications*, volume 1, pages 464–471, 2009. 3
- [DVP00] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. *Conference on Computer Vision and Pattern Recognition*, 2 :142–149, 2000. 7, 14, 20
- [FBLF08] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-Camera People Tracking with a Probabilistic Occupancy Map. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2) :267–282, 2008. 7, 10, 13, 15, 20
- [GB06] H. Grabner and H. Bischof. On-line boosting and vision. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 260–267, Washington, DC, USA, 2006. IEEE Computer Society. 6

- [GCMT06] Yann Goyat, Thierry Chateau, Laurent Malaterre, and Laurent Trassoudaine. Vehicle trajectories evaluation by static video sensors. In *9th International IEEE Intelligent Transportation Systems Conference (ITSC 2006)*, 2006. 92
- [GCT09] Y. Goyat, T. Chateau, and L. Trassoudaine. Tracking of vehicle trajectory by combining a camera and a laser rangefinder. *Machine Vision and Applications*, 2009. 13
- [Gen92] D.B. Gennery. Visual tracking of known three-dimensional objects. *International Journal of Computer Vision*, 7(3) :243–270, April 1992. 11
- [Gil92] T. Gillespie. *Fundamentals of Vehicle Dynamics*. Society of Automotive Engineers, 1992. 92
- [GM07] D. M. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *Int. J. Comput. Vision*, 73(1) :41–59, 2007. 14
- [Gre95] Peter J. Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 4(82) :711–732, 1995. 68, 71
- [IB98] M. Isard and A. Blake. A mixed-state condensation tracker with automatic model-switching. In *Computer Vision, 1998. Sixth International Conference on*, pages 107–112, 1998. 4
- [IM01] Michael Isard and John MacCormick. Bramble : A bayesian multiple-blob tracker. In *Proc. Int. Conf. Computer Vision*, vol. 2 34-41, 2001. 3, 26, 27, 30
- [JD01] Frédéric Jurie and Michel Dhome. Real time 3d template matching. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1 :791, 2001. 7
- [KB05] S. J. Kanhere, N. K. Pundlik and S. T. Birchfield. Vehicle segmentation and tracking from a low-angle off-axis camera. In *CVPR, Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1152–1157, 2005. 3, 13
- [KBD04] Z. Khan, T. Balch, and F. Dellaert. An MCMC-based particle filter for tracking multiple interacting targets. *ECCV*, 3024 :279–290, 2004. 19, 21, 26, 29, 42, 68
- [KBD05] Z. Khan, T. Balch, and F. Dellaert. MCMC-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27 :1805 – 1918, 2005. 3, 26, 41, 42, 68, 71, 73, 83, 90, 102, 105, 106, 107, 129
- [KCF06] Bernd Roessler Kay Ch. Fuerstemberg. Advanced intersection safety - the ec project intersafe. In *International IEEE Intelligent Vehicle Symposium*, pages 89–93, 2006. 3
- [KDN93] D. Koller, K. Daniilidis, and H. H. Nagel. Model-based object tracking in monocular image sequences of road traffic scenes. *International Journal of Computer Vision*, 10(3) :257–281, 1993. 3
- [LBCT03] J. Laneurit, C. Blanc, R. Chapuis, and L. Trassoudaine. Multisensorial data fusion for global vehicle and obstacles absolute positioning. In *Intelligent Vehicle Symposium*, pages 138–143, 2003. 3
- [LCCVG07] B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool. Dynamic 3d scene analysis from a moving vehicle. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007. 14
- [LSCG08] Bastian Leibe, Konrad Schindler, Nico Cornelis, and Luc Van Gool. Coupled object detection and tracking from static cameras and moving vehicles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10) :1683–1698, 2008. 10

- [LSS] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, pages 878–885 vol. 1. 10
- [MA98] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *IJCV : International Journal of Computer Vision*, 29(1) :5–28, 1998. 3, 11, 15, 26, 29, 71
- [Mac03] David MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003. 29, 68
- [MB99] John MacCormick and Andrew Blake. A probabilistic exclusion principle for tracking multiple objects. In *Int. Conf. Computer Vision*, 572–578, 1999. 15, 26, 30, 34
- [MJL06] Joseph L. Mundy Matthew J. Leotta. Learning background and shadow appearance with 3-d vehicle models. In *British Machine Vision Conference (BMVC)*, volume 2, pages 649–658, september 2006. 16
- [MR04] Antoine Manzanera and Julien Richefeu. A robust and computationally efficient motion detection algorithm based on sigma-delta background estimation. In *Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP'04)*, pages 46–51, 2004. 15, 79
- [ÖLFF06] Mustafa Özuysal, Vincent Lepetit, François Fleuret, and Pascal Fua. Feature harvesting for tracking-by-detection. In *ECCV (3)*, pages 592–605, 2006. 8
- [OTF<sup>+</sup>04] Kenji Okuma, Ali Taleghani, Nando De Freitas, O De Freitas, James J. Little, and David G. Lowe. A boosted particle filter : Multitarget detection and tracking. In *In ECCV*, pages 28–39, 2004. 14
- [OWWY06] T. Osawa, X.J. Wu, K. Wakabayashi, and T. Yasuno. Human tracking by particle filtering using full 3d model of both target and environment. pages II : 25–28, 2006. 20
- [PCJM02] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-Based Probabilistic Tracking. In *Computer Vision ECCV 2002*, volume 1, pages 661–675, May 2002. 14, 20, 130
- [PMT03] Andrea Prati, Ivana Mikic, Mohan M. Trivedi, and Rita Cucchiara. Detecting moving shadows : Algorithms and evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25 :918–923, 2003. 15
- [RSW08] E. Richter, R. Schubert, and G. Wanielik. Radar and vision based data fusion - advanced filtering techniques for a multi object vehicle tracking system. In *Intelligent Vehicle Symposium*, pages 120–125, 2008. 3
- [Rub88] D. B. Rubin. Using the SIR algorithm to simulate posterior distributions. In M. H. Bernardo, K. M. Degroot, D. V. Lindley, and A. F. M. Smith, editors, *Bayesian Statistics 3*. Oxford University Press, 1988. 26
- [SCE04] E. Salvador, A. Cavallaro, and T. Ebrahimi. Cast shadow segmentation using invariant color features. *Computer Vision and Image Understanding*, 95(2) :238 – 259, August 2004. 15
- [SGP04] Kevin Smith and Daniel Gatica-Perez. Order matters : A distributed sampling method for multi-object tracking. In *British Machine Vision Conference (BMVC)*, 2004. 30, 34
- [Smi07] Kevin Smith. *Bayesian Methods for Visual Multi-Object Tracking with Applications to Human Activity Recognition*. PhD thesis, EPFL, Lausanne, Suisse, 2007. 26, 37
- [Sta99] W.E.L. Stauffer, C. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition*, volume 2, 1999. 15

- [VJ01] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1 :511, 2001. 6
- [WN07] Bo Wu and ram Nevatia. Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *International Journal of Computer Vision*, 75(2) :247–266, November 2007. 14
- [Wre97] A. Darrell T. Pentland A. Wren, C. Azarbayejani. Pfunder : Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 19(7), 1997. 15
- [YGBG03] Danny B. Yang, Héctor H. González-baños, and Leonidas J. Guibas. Counting people in crowds with a real-time network of simple image sensors. In *In Proc. IEEE International Conference on Computer Vision*, pages 122–129, 2003. 6
- [YMC07] Q. Yu, G. Medioni, and I. Cohen. Multiple target tracking using spatio-temporal markov chain monte carlo data association. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1 – 8, 2007. 9, 36
- [YO08] Jian Yao and Jean-Marc Odobez. Multi-camera multi-person 3D space tracking with mcmc in surveillance scenarios. In *European Conference on Computer Visionworkshop on Multi Camera and Multi-modal Sensor Fusion Algorithms and Applications (ECCV-M2SFA2)*, 2008. 20, 90
- [ZMR<sup>+</sup>08] Beibei Zhan, Dorothy N. Monekosso, Paolo Remagnino, Sergio A. Velastin, and Li-Qun Xu. Crowd analysis : a survey. *Machine Vision and Applications*, 19(5-6) :345–357, 2008. 3
- [ZN04] Tao Zhao and Ram Nevatia. Tracking multiple humans in crowded environment. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2 :406–413, 2004. 20, 21

# Résumé

Cette thèse présente une méthode de suivi et de classification conjoints en temps réel d'un nombre variable d'objets tels que des piétons et / ou des véhicules, sous conditions d'illumination variables au cours du temps. Le système a été développé au LASMEA<sup>1</sup> de 2006 à 2009, sous la direction de Michel Naranjo et l'encadrement de Thierry Chateau. Nous posons d'abord un panorama des applications nécessitant le suivi et la catégorisation d'objets en temps réel, les problématiques liées au suivi en temps réel de multiples objets, et les méthodes couramment employées. La méthode que nous avons retenue entre dans le champ du suivi Multi-Objets par Filtre Particulaire, dont la clé de voûte est l'échantillonnage des particules. Après avoir rappelé le principe des Filtres Particulaires, nous nous focalisons sur la question du rééchantillonnage des particules. Nous examinons deux familles de filtres particuliers permettant d'opérer dans un espace de grande dimension : les Filtres Particulaires Partitionnés, et les Filtres Particulaires par Chaîne de Markov (*FP MCMC*). Nous comparons ensuite leurs performances sur des données de synthèse. Les résultats obtenus montrent la supériorité du Filtre Particulaire MCMC (méthode de Monte-Carlo par Chaîne de Markov) à mouvements marginalisés. Un système de suivi et classification conjoints en temps réel d'un nombre variable d'objets tels que des piétons et/ou des véhicules, sous illumination variable, est ensuite présenté. La mesure est délivrée par une ou plusieurs caméras statiques. Nous avons délibérément choisi d'alimenter le filtre avec une observation pauvre, reposant uniquement sur une segmentation binaire *avant-plan / arrière-plan* basée sur un modèle de l'*arrière-plan* mis à jour en ligne à chaque image. Pour résister aux variations d'illumination, les ombres sont modélisées et le filtre est étendu afin de suivre conjointement le soleil et les objets au moyen du (*FP MCMC*). Les résultats de suivi et classification en temps réel sont présentés et discutés sur des séquences réelles et sur des séquences de synthèse, impliquant plusieurs catégories d'utilisateurs tels que des piétons, des voitures, des camionnettes et des poids lourds.

**Mots Clef** Suivi Multi-Objets, Suivi Visuel, Temps Réel, Filtre Particulaire MCMC.

---

<sup>1</sup>Laboratoire des Sciences et Matériaux pour l'Électronique et l'Automatique, UMR CNRS - Université Blaise-Pascal